

<project name="seance05" app="Snap! 6, https://snap.berkeley.edu" version="1"><notes></notes><thumbnail>  
Ly4Ha74XQ6MTIygubmZpw7dw5ppqfMxo2KtJdDQwdWYtvdnwDp1G5Z8pLTmv83Xhz9aj6mLDZZglnTlnfC5TsPv7oV99RZg1Aej7yaGjxbAU7APV+Izrf8E5hw50TkRt1EcKDQ7YGtqajL8fn8A/pgED/s5I2k6Mr56LvITbD8N1+/  
DlnwNDQ0oKysLfgZXjMgdsJnn36amJnR0dGDBggWTGvLv7BM6g0THGSK193djVu3bmH37t3BWJjrcnKBS0Ahs3LMKNwBChiFUCVvooCSaEXhrBQwCqFKqkQBjDgKwlnkFPDMmT0Y03cu7HZ7U0qbX/ncvn0b5tUVPrgD4wpYXV2NzMxM6/s58zH29cu/n99qrkXS2fcRP2X8DZ05Pw0Dm7/A60gozJsP1qxZQwKa78C4Al64cGFS2/  
Nb2fMoTPMGjrU5EmF/cidiU5fC2/  
KddQV7k7NG19Qf4fD7r6cDAQ0Dqy6SCeFDU7cC4ArpcLquweVmusrISGzZsgMPhsE7HMTExMK8Xmz/  
bPyzC02n3pXK8VglbQhKMwR7ELloNf99NjFQ8uHrS8XJNQMCrV6+iuLg46jaVhSa/A+MKeP78eWulr4uHDlyBCULJViyZAlmz55t3bd34sQJS6b0j19C4V8CmsfHpCxGf0E7iHlkuXUTgveXbwMTXS4+HRCwr68PBQUfK5+WR0bdDowr4MmTJ5Gebt5gcP/3v7Hf/bxeL0x5zJsX2tvbMaWxDMS8FyfenLgEtBV8bgbk40Dho3WUTFxc38ft4RNTuwLgCDg0NoaKiivrUS0hI+N9NMF8zT80TPTweD/r7+63rzkvFRUhJSZnoLXw9yndgwq9horw/6yneAQqoGIDu8RRQdwMU96eAigHoHk8BdTdAcX8KqBiA7vEUUHcDFPengIoB6B5PAXU3QHF/  
CqgYg07xFFB3AxT3p4CKAegeTwF1N0BxfwqoGIDu8RRQdwMU96eAigHoHk8BdTdAcX8KqBiA7vEUUHcDFPengIoB6B5PAXU3QHF/  
CqgYg07xFFB3AxT3p4CKAegeTwF1N0BxfwqoGIDu8RRQdwMU96eAigHoHk8BdTdAcX8KqBiA7vEUUHcDFPengIoB6B5PAXU3QHF/  
CqgYg07xFFB3AxT3p4CKAegeTwF1N0BxfwqoGIDu8RRQdwMU96eAigHoHk8BdTdAcX8KqBiA7vEUUHcDFPengIoB6B5PAXU3QHF/  
CqgYg07xFFB3AxT3p4CKAegeTwF1N0BxfwqoGIDu8RQwjAY8syoLNc0XwpgY+VEUMEyMDMPAsWUx2PWrEaZEGTEUMEycGutqcGnPs3Ds+ATbX98RptTIj6GAYWL03nInEn1DuGvEIDb0Ab/Pi2ZPEqou94ZpgsiMoYBh40IeHEBlbtI/khzbDmP7rjfCk8B7ZERQwDhw2Ls9Anu+aLeSHDa/+1I2Zc5LDkBz5ERQwxIz8fj8qsm0tLL64GXjrfA/sdnuIU+UsTwFDz0r7qm9wY/9mXEp/Ap/WnA1xmrzLkWCImR1a0R25H5zC2nX5IU6SuTwFDcG3Ua8HI/c8cDqdIUyRvTQFLM1P/PQUUDx2QUooGx+4qeng0IRyi5AAWxzEz89BRSPUHYBCiibn/jpKaB4hLILUEDZ/MRPTwHFI5RdgALK5id+egooHqHsAhRQNj/x01NA8Qhlf6CAsvmJn54CikcouwAF LM1P/PQUUDx2QUooGx+4qeng0IRyi5AAWxzEz89BRSPUHYBCiibn/jpKaB4hLILUEDZ/MRPTwHFI5RdgALK5id+egooHqHsAhRQNj/x01NA8Qhlf6CAsvmJn54CikcouwAF LM1P/PQUUDx2QUooGx+4qeng0IRyi5AAWxzEz89BRSPUHYBCiibn/jpKaB4hLILUEDZ/MRP/ydWjYDGPw+QEGAAAABJRu5ErkJggg==</thumbnail><stage name="Stage" width="480" height="360" costume="0" color="255,255,255,1" tempo="60" threadsafe="false" penlog="false" volume="100" pan="0" lines="round" ternary="true" hyperops="true" codify="false" inheritance="false" sublistIDs="false" scheduled="false"

id="1"><pentrails>  
u3VwQkAAAjEMN1/  
abewn7jAQRc64wgQIECAAIIF3gX1fNEiAAAEcBAiMAHsCAgQIECAQCAhwgG6SAAECBAgI  
sB8gQIAAAQKBgAAH6CYJECBAgIAA+wECBAgQIBAICHCAbpIAAQIECAiwHyBAgAABAoGA  
AAfoJgkQIECAGAD7AQIECBAgEAgIcIBukgABAgQICLAFIECAAECgYAAB+gmCRAGQICA  
APsBAgQIECAQCAhwgG6SAAECBAgIsB8gQIAAAQKBgAAH6CYJECBAgIAA+wECBAgQIBA  
ICHCAbpIAAQIECAiwHyBAgAABAoGAAAfoJgkQIECAGAD7AQIECBAgEAgIcIBukgABAgQI  
CLAFIECAAECgYAAB+gmCRAGQICAAPsBAgQIECAQCAhwgG6SAAECBAgIsB8gQIAAAQKB  
gAAH6CYJECBAgIAA+wECBAgQIBAICHCAbpIAAQIECAiwHyBAgAABAoGAAAfoJgkQIECA  
gAD7AQIECBAgEAgIcIBukgABAgQICLAFIECAAECgYAAB+gmCRAGQICAAPsBAgQIECAQ  
CAhwgG6SAAECBAgIsB8gQIAAAQKBgAAH6CYJECBAgIAA+wECBAgQIBAICHCAbpIAAQIE  
CAiwHyBAgAABAoGAAAfoJgkQIECAGAD7AQIECBAgEAgIcIBukgABAgQICLAFIECAAEC  
gYAAB+gmCRAGQICAAPsBAgQIECAQCAhwgG6SAAECBAgIsB8gQIAAAQKBgAAH6CYJECBA  
gIAA+wECBAgQIBAICHCAbpIAAQIECAiwHyBAgAABAoGAAAfoJgkQIECAGAD7AQIECBAg  
EAgIcIBukgABAgQICLAFIECAAECgYAAB+gmCRAGQICAAPsBAgQIECAQCAhwgG6SAAEC  
BAgIsB8gQIAAAQKBgAAH6CYJECBAgIAA+wECBAgQIBAICHCAbpIAAQIECAiwHyBAgAAB  
AoGAAAfoJgkQIECAGAD7AQIECBAgEAgIcIBukgABAgQICLAFIECAAECgYAAB+gmCRAG  
QICAAPsBAgQIECAQCAhwgG6SAAECBAgIsB8gQIAAAQKBgAAH6CYJECBAgIAA+wECBAgQ  
IBAICHCAbpIAAQIECAiwHyBAgAABAoGAAAfoJgkQIECAGAD7AQIECBAgEAgIcIBukgAB  
AgQICLAFIECAAECgYAAB+gmCRAGQICAAPsBAgQIECAQCAhwgG6SAAECBAgIsB8gQIAA  
AQKBgAAH6CYJECBAgIAA+wECBAgQIBAICHCAbpIAAQIECAiwHyBAgAABAoGAAAfoJgkQ  
IECAGAD7AQIECBAgEAgIcIBukgABAgQICLAFIECAAECgYAAB+gmCRAGQICAAPsBAgQI  
ECAQCAhwgG6SAAECBAgIsB8gQIAAAQKBgAAH6CYJECBAgIAA+wECBAgQIBAICHCAbpIA  
AQIECAiwHyBAgAABAoGAAAfoJgkQIECAGAD7AQIECBAgEAgIcIBukgABAgQICLAFIECA  
AAECgYAAB+gmCRAGQICAAPsBAgQIECAQCAhwgG6SAAECBAgIsB8gQIAAAQKBgAAH6CYJ  
ECBAgIAA+wECBAgQIBAICHCAbpIAAQIECAiwHyBAgAABAoGAAAfoJgkQIECAGAD7AQIE  
CBAgEAgIcIBukgABAgQICLAFIECAAECgYAAB+gmCRAGQICAAPsBAgQIECAQCAhwgG6S  
AAECBAgIsB8gQIAAAQKBgAAH6CYJECBAgIAA+wECBAgQIBAICHCAbpIAAQIECAiwHyBA  
gAABAoGAAAfoJgkQIECAGAD7AQIECBAgEAgIcIBukgABAgQICLAFIECAAECgYAAB+gm  
CRAGQICAAPsBAgQIECAQCAhwgG6SAAECBAgIsB8gQIAAAQKBgAAH6CYJECBAgIAA+wEC  
BAgQIBAICHCAbpIAAQIECAiwHyBAgAABAoGAAAfoJgkQIECAGAD7AQIECBAgEAgIcIBu  
kgABAgQICLAFIECAAECgYAAB+gmCRAGQICAAPsBAgQIECAQCAhwgG6SAAECBAgIsB8g  
QIAAAQKBgAAH6CYJECBAgIAA+wECBAgQIBAICHCAbpIAAQIECAiwHyBAgAABAoGAAAfo  
JgkQIECAGAD7AQIECBAgEAgIcIBukgABAgQICLAFIECAAECgYAAB+gmCRAGQICAAPsB  
AgQIECAQCAhwgG6SAAECBAgIsB8gQIAAAQKBgAAH6CYJECBAgIAA+wECBAgQIBAICHCA  
bpIAAQIECAiwHyBAgAABAoGAAAfoJgkQIECAGAD7AQIECBAgEAgIcIBukgABAgQICLAF  
IECAAECgYAAB+gmCRAGQICAAPsBAgQIECAQCAhwgG6SAAECBAgIsB8gQIAAAQKBgAAH  
6CYJECBAgIAA+wECBAgQIBAICHCAbpIAAQIECAiwHyBAgAABAoGAAAfoJgkQIECAGAD7  
AQIECBAgEAgIcIBukgABAgQICLAFIECAAECgYAAB+gmCRAGQICAAPsBAgQIECAQCAhw  
gG6SAAECBAgIsB8gQIAAAQKBgAAH6CYJECBAgIAA+wECBAgQIBAICHCAbpIAAQIECAiw  
HyBAgAABAoGAAAfoJgkQIECAGAD7AQIECBAgEAgIcIBukgABAgQICLAFIECAAECgYAA  
B+gmCRAGQICAAPsBAgQIECAQCAhwgG6SAAECBAgIsB8gQIAAAQKBgAAH6CYJECBAgIAA  
+wECBAgQIBAICHCAbpIAAQIECAiwHyBAgAABAoGAAAfoJgkQIECAGAD7AQIECBAgEAgI  
cIBukgABAgQICLAFIECAAECgYAAB+gmCRAGQICAAPsBAgQIECAQCAhwgG6SAAECBAgI  
sB8gQIAAAQKBgAAH6CYJECBAgIAA+wECBAgQIBAICHCAbpIAAQIECAiwHyBAgAABAoGA  
AAfoJgkQIECAGAD7AQIECBAgEAgIcIBukgABAgQICLAFIECAAECgYAAB+gmCRAGQICA  
APsBAgQIECAQCAhwgG6SAAECBAgIsB8gQIAAAQKBgAAH6CYJECBAgIAA+wECBAgQIBA  
ICHCAbpIAAQIECAiwHyBAgAABAoGAAAfoJgkQIECAGAD7AQIECBAgEAgIcIBukgABAgQI  
CLAFIECAAECgYAAB+gmCRAGQICAAPsBAgQIECAQCAhwgG6SAAECBAgIsB8gQIAAAQKB  
gAAH6CYJECBAgIAA+wECBAgQIBAICHCAbpIAAQIECAiwHyBAgAABAoGAAAfoJgkQIECA  
gAD7AQIECBAgEAgIcIBukgABAgQICLAFIECAAECgYAAB+gmCRAGQICAAPsBAgQIECAQ  
CAhwgG6SAAECBAgIsB8gQIAAAQKBgAAH6CYJECBAgIAA+wECBAgQIBAICHCAbpIAAQIE  
CAiwHyBAgAABAoGAAAfoJgkQIECAGAD7AQIECBAgEAgIcIBukgABAgQICLAFIECAAEC

gYAAB+gmCRAGQICAAPsBAGQIECAQCAhwgG6SAAECBAGIsB8gQIAAAQKBgAAH6CYJECBA  
gIAA+wECBAGQIBAICHCAbpIAAQIECAiwHyBAGAABoGAAAfoJgkQIECAGAD7AQIECBAG  
EAgIcIBukgABAGQICLAFIECAAAECgYAAB+gmCRAGQICAAPsBAGQIECAQCAhwgG6SAAEC  
BAGIsB8gQIAAAQKBgAAH6CYJECBAgIAA+wECBAGQIBAICHCAbpIAAQIECAiwHyBAGAA  
BoGAAAfoJgkQIECAGAD7AQIECBAGEAgIcIBukgABAGQICLAFIECAAAECgYAAB+gmCRAG  
QICAAPsBAGQIECAQCAhwgG6SAAECBAGIsB8gQIAAAQKBgAAH6CYJECBAgIAA+wECBAGQ  
IBAICHCAbpIAAQIECAiwHyBAGAABoGAAAfoJgkQIECAGAD7AQIECBAGEAgIcIBukgAB  
AgQICLAFIECAAAECgYAAB+gmCRAGQICAAPsBAGQIECAQCAhwgG6SAAECBAGIsB8gQIAA  
AQKBgAAH6CYJECBAgIAA+wECBAGQIBAICHCAbpIAAQIECAiwHyBAGAABoGAAAfoJgkQ  
IECAGAD7AQIECBAGEAgIcIBukgABAGQICLAFIECAAAECgYAAB+gmCRAGQICAAPsBAGQI  
ECAQCAhwgG6SAAECBAGIsB8gQIAAAQKBgAAH6CYJECBAgIAA+wECBAGQIBAICHCAbpIA  
AQIECAiwHyBAGAABoGAAAfoJgkQIECAGAD7AQIECBAGEAgIcIBukgABAGQICLAFIECA  
AAECgYAAB+gmCRAGQICAAPsBAGQIECAQCAhwgG6SAAECBAGIsB8gQIAAAQKBgAAH6CYJ  
ECBAgIAA+wECBAGQIBAICHCAbpIAAQIECAiwHyBAGAABoGAAAfoJgkQIECAGAD7AQIE  
CBAGEAgIcIBukgABAGQICLAFIECAAAECgYAAB+gmCRAGQICAAPsBAGQIECAQCAhwgG6S  
AAECBAGIsB8gQIAAAQKBgAAH6CYJECBAgIAA+wECBAGQIBAICHCAbpIAAQIECAiwHyBA  
gAABoGAAAfoJgkQIECAGAD7AQIECBAGEAgIcIBukgABAGQICLAFIECAAAECgYAAB+gm  
CRAGQICAAPsBAGQIECAQCAhwgG6SAAECBAGIsB8gQIAAAQKBgAAH6CYJECBAgIAA+wEC  
BAGQIBAICHCAbpIAAQIECAiwHyBAGAABoGAAAfoJgkQIECAGAD7AQIECBAGEAgIcIBu  
kgABAGQICLAFIECAAAECgYAAB+gmCRAGQICAAPsBAGQIECAQCAhwgG6SAAECBAGIsB8g  
QIAAAQKBgAAH6CYJECBAgIAA+wECBAGQIBAICHCAbpIAAQIECAiwHyBAGAABoGAAAfo  
JgkQIECAGAD7AQIECBAGEAgIcIBukgABAGQIHLFXAWmHEWHPAAAAAEFLTksuQmCC</  
pentrails><costumes><list struct="atomic" id="2"></list></  
costumes><sounds><list struct="atomic" id="3"></list></  
sounds><variables></variables><blocks></blocks><scripts></  
scripts><sprites><sprite name="Sprite" idx="1" x="23.9152542372874"  
y="-55.11864406779671" heading="217" scale="1" volume="100" pan="0"  
rotation="1" draggable="true" costume="0"  
color="145.35,43.605000000000004,0,1" pen="tip"  
id="8"><costumes><list struct="atomic" id="9"></list></  
costumes><sounds><list struct="atomic" id="10"></list></  
sounds><blocks></blocks><variables></variables><scripts><script  
x="30" y="20"><custom-block s="set \$robot host to  
%s"><l>192.168.0.33</l></custom-block></script><script  
x="32.53333282470703" y="59.00000200000002"><custom-block s="\$robot  
test connection"></custom-block></script><script x="87"  
y="125"><block s="doForever"><script><custom-block s="\$robot set  
position(s) %n of motor(s) %s in %n seconds | wait ? %boolUE"><block  
var="ang"/><l>m6</l><l>0</l></custom-block><block  
s="doIfElse"><block s="reportGreaterThan"><block var="ang"/><l>0</  
l></block><script><block s="errorObsolete"></block></  
script><script><block s="errorObsolete"></block></script></block></  
script></block></script><script x="282.99999999999994" y="30"><block  
s="doForever"><script><block s="doSetVar"><l>ang</l><block  
s="reportDifference"><block s="reportQuotient"><block  
s="errorObsolete"></block><l>10</l></block><l>50</l></block></  
block></script></block></script></scripts></sprite><watcher  
var="port" style="normal" x="0.0000029305615498742554"  
y="0.000009084741861897783" color="243,118,29" hidden="true"/  
><watcher var="host" style="normal" x="0.0005515317122899432"  
y="0.006754944721862444" color="243,118,29"/><watcher var="ang"  
style="normal" x="19.6666666666666742" y="36.260416666666666"  
color="243,118,29"/></sprites></stage><hidden></hidden><headers></  
headers><code><delim></delim><string>&lt;#1&gt;</

```

string><parms_list>Context &gt;&gt; a CommandBlockMorph
(&quot;script variables %scriptVars...&quot;) a VariableFrame {}</
parms_list><tempvars_delim>Context &gt;&gt; a CommandBlockMorph
(&quot;script variables %scriptVars...&quot;) a VariableFrame {}</
tempvars_delim><doSetVar>Context &gt;&gt; a CommandBlockMorph
(&quot;script variables %scriptVars...&quot;) a VariableFrame {}</
doSetVar></code><blocks><block-definition s="$robot motors in group
%&apos;alias&apos;" type="reporter" category="sensing"><comment
x="0" y="0" w="302" collapsed="false">Return a list of motors which
are grouped in the input alias</comment><header></header><code></
code><translations></translations><inputs><input
type="%txt">group_name</input></inputs><script><block
s="doDeclareVariables"><list><l>output</l></list></block><block
s="doSetVar"><l>output</l><block s="reportURL"><block
s="reportJoinWords"><list><custom-block s="$robot robot URL"></
custom-block><l>/motors/</l><block var="alias"/></list></block></
block></block><block s="doIfElse"><block s="reportEquals"><block
var="output"/><l></l></block><script><custom-block s="$robot popup
%txt"><block s="reportJoinWords"><list><l>Group &lt; </l><block
var="alias"/><l> &gt; is not in the available groups of your robot :
</l><custom-block s="combine with %repRing items of %l"><block
s="reifyReporter"><autolambda><custom-block s="join words
%mult%txt"><list><l></l><l></l></list></custom-block></
autolambda><list></list></block><custom-block s="$robot all motors
groups"></custom-block></custom-block></list></block></custom-
block><block s="doReport"><l></l></block></script><script><block
s="doReport"><block s="reportTextSplit"><block var="output"/><l></l></
block></block></script></block></script></block-
definition><block-definition s="$robot all motors" type="reporter"
category="sensing"><comment x="0" y="0" w="90"
collapsed="false">Return a list of all available motors</
comment><header></header><code></code><translations></
translations><inputs></inputs><script><block s="doReport"><custom-
block s="$robot motors in group %txt"><l>motors</l></custom-block></
block></script></block-definition><block-definition s="$robot get
all motors positions" type="reporter" category="sensing"><comment
x="0" y="0" w="170" collapsed="false">Return a list of all motors
present positions, in the order of &quot;$robot all motors&quot;
block list.&#xD;&#xD;Usefull for some advanced programs.</
comment><header></header><code></code><translations></
translations><inputs></inputs><script><block s="doReport"><block
s="reportTextSplit"><block s="reportURL"><block
s="reportJoinWords"><list><custom-block s="$robot robot URL"></
custom-block><l>/motors/get/positions</l></list></block></
block><l>;</l></block></block></script></block-definition><block-
definition s="if %&apos;test&apos; then %&apos;true&apos; else
%&apos;false&apos;" type="reporter" category="control"><header></
header><code></code><translations></translations><inputs><input
type="%b"></input><input type="%anyUE"></input><input
type="%anyUE"></input></inputs><script><block s="doIfElse"><block
var="test"/><script><block s="doReport"><block s="evaluate"><block
var="true"/><list></list></block></block></script><script><block
s="doReport"><block s="evaluate"><block var="false"/><list></list></
block></block></script></block></script></block-definition><block-

```

```

definition s="for %&apos;i&apos; = %&apos;start&apos; to
%&apos;end&apos; %&apos;action&apos;" type="command"
category="control"><header></header><code></code><translations></
translations><inputs><input type="%upvar"></input><input
type="%n">1</input><input type="%n">10</input><input type="%cs"></
input></inputs><script><block s="doDeclareVariables"><list><l>step</
l><l>tester</l></list></block><block s="doIfElse"><block
s="reportGreaterThan"><block var="start"/><block var="end"/></
block><script><block s="doSetVar"><l>step</l><l>-1</l></block><block
s="doSetVar"><l>tester</l><block
s="reifyReporter"><autolambda><block s="reportLessThan"><block
var="i"/><block var="end"/></block></autolambda><list></list></
block></block></script><script><block s="doSetVar"><l>step</l><l>1</
l></block><block s="doSetVar"><l>tester</l><block
s="reifyReporter"><autolambda><block s="reportGreaterThan"><block
var="i"/><block var="end"/></block></autolambda><list></list></
block></block></script></block><block s="doSetVar"><l>i</l><block
var="start"/></block><block s="doUntil"><block s="evaluate"><block
var="tester"/><list></list></block><script><block s="doRun"><block
var="action"/><list></list></block><block s="doChangeVar"><l>i</
l><block var="step"/></block></script></block></script></block-
definition><block-definition s="$robot get %&apos;register&apos; of
motor(s) %&apos;motors&apos;" type="reporter"
category="sensing"><comment x="0" y="0" w="230.00000099999988"
collapsed="false">Get a defined &lt;register&gt; &lt;value&gt; of
one motor.&#xD;Example of some registers : torque_limit,
angle_limit, present_temperature, present_position...&#xD;For more
informations about registers, look at the online documentation :
http://$robot-project.github.io/pybot/&#xD;</comment><header></
header><code></code><translations></translations><inputs><input
type="%s">present_position<options>position=present_position&#xD;spe
ed=present_speed&#xD;load
(force)=present_load&#xD;temperature=present_temperature&#xD;voltage
=present_voltage</options></input><input type="%s">motor_name</
input></inputs><script><block
s="doDeclareVariables"><list><l>output</l><l>complete motor list</
l></list></block><block s="doSetVar"><l>complete motor list</
l><custom-block s="$robot all motors"></custom-block></block><block
s="doIf"><block s="reportNot"><block s="reportIsA"><block
var="motors"/><l><option>list</option></l></block></
block><script><block s="doIfElse"><block s="reportOr"><block
s="reportIsIdentical"><block var="motors"/><l>all</l></block><block
s="reportIsIdentical"><block var="motors"/><l></l></block></
block><script><block s="doSetVar"><l>motors</l><custom-block
s="$robot all motors"></custom-block></block></script><script><block
s="doSetVar"><l>motors</l><block s="reportTextSplit"><block
var="motors"/><l><option>whitespace</option></l></block></block></
script></block></script></block><custom-block s="for each %upvar of
%l %cs"><l>m</l><block var="motors"/><script><block s="doIf"><block
s="reportNot"><block s="reportListContainsItem"><block var="complete
motor list"/><block var="m"/></block></block><script><custom-block
s="$robot popup %txt"><block s="reportJoinWords"><list><l>Motor &lt;
</l><block var="m"/><l> &gt; is not in the available motors of your
robot : </l><custom-block s="combine with %repRing items of

```

```

%l"><block s="reifyReporter"><autolambda><custom-block s="join words
%mult%txt"><list><l></l><l></l></list></custom-block></
autolambda><list></list></block><block var="complete motor list"/></
custom-block></list></block></custom-block><block s="doReport"><l></
l></block></script></block></script></custom-block><block
s="doIfElse"><block s="reportEquals"><block
s="reportListLength"><block var="motors"/></block><l>1</l></
block><script><block s="doReport"><block s="reportURL"><block
s="reportJoinWords"><list><custom-block s="$robot robot URL"></
custom-block><l>/motors/</l><custom-block s="combine with %repRing
items of %l"><block s="reifyReporter"><autolambda><block
s="reportJoinWords"><list><block var="#1"/><l>;</l><block var="#2"/
></list></block></autolambda><list><l>#1</l><l>#2</l></list></
block><block var="motors"/></custom-block><l>/get/</l><block
var="register"/></list></block></block></block></
script><script><block s="doReport"><block s="reportTextSplit"><block
s="reportURL"><block s="reportJoinWords"><list><custom-block
s="$robot robot URL"></custom-block><l>/motors/</l><custom-block
s="combine with %repRing items of %l"><block
s="reifyReporter"><autolambda><block
s="reportJoinWords"><list><block var="#1"/><l>;</l><block var="#2"/
></list></block></autolambda><list><l>#1</l><l>#2</l></list></
block><block var="motors"/></custom-block><l>/get/</l><block
var="register"/></list></block></block><l>;</l></block></block></
script></block></script></block-definition><block-definition
s="$robot index of motor 'motor'" type="reporter"
category="sensing"><comment x="0" y="0" w="109.00000000000001"
collapsed="false">Return the index of a motor name in the '
$robot all motors' block list</comment><header></
header><code></code><translations></translations><inputs><input
type="%txt">motor_name</input></inputs><script><block
s="doDeclareVariables"><list><l>i</l><l>complete motor list</l></
list></block><block s="doSetVar"><l>complete motor list</l><custom-
block s="$robot all motors"></custom-block></block><block
s="doSetVar"><l>i</l><l>1</l></block><custom-block s="for each
%upvar of %l %cs"><l>m</l><block var="complete motor list"/
><script><block s="doIf"><block s="reportEquals"><block var="m"/
><block var="motor"/></block><script><block s="doReport"><block
var="i"/></block></script></block><block s="doChangeVar"><l>i</
l><l>1</l></block></script></custom-block><custom-block s="$robot
popup %txt"><block s="reportJoinWords"><list><l>Motor &lt; </
l><block var="motor"/><l>&gt; is not in the available motors of your
robot : </l><custom-block s="combine with %repRing items of
%l"><block s="reifyReporter"><autolambda><custom-block s="join words
%mult%txt"><list><l></l><l></l></list></custom-block></
autolambda><list></list></block><block var="complete motor list"/></
custom-block></list></block></custom-block><block s="doReport"><l></
l></block></script></block-definition><block-definition s="$robot
robot URL" type="reporter" category="sensing"><comment x="0" y="0"
w="117" collapsed="false">For internal use only</comment><header></
header><code></code><translations></translations><inputs></
inputs><script><block s="doReport"><block
s="reportJoinWords"><list><block var="host"/><l>:</l><block
var="port"/></list></block></block></script></block-

```

```

definition><block-definition s="$robot set motor(s)
'motor'; 'compliant';" type="command"
category="control"><comment x="0" y="0" w="230.00000099999988"
collapsed="false">Compliance property make motors actively become
soft and movable.&#xD;You need to activate the compliance if you
want to record a movement by demonstration.&#xD;&#xD;The motor(s)
input can accept :&#xD;- string of a motor name&#xD;- string of many
motors separed with spaces&#xD;- a Snap! list of motors like the
reporter block &quot;$robot all motors&quot;</comment><header></
header><code></code><translations></translations><inputs><input
type="%s">motor_name</input><input type="%txt"
readonly="true">compliant<options>compliant&#xD;stiff</options></
input></inputs><script><block s="doDeclareVariables"><list><l>comp</
l></list></block><block s="doSetVar"><l>comp</l><custom-block s="if
%b then %anyUE else %anyUE"><block s="report0r"><block
s="reportEquals"><block var="compliant"/><l>compliant</l></
block><block var="compliant"/></block><l>1</l><l>0</l></custom-
block></block><custom-block s="$robot set %txt of motor(s) %s to
%s"><l>compliant</l><block var="motor"/><block var="comp"/></custom-
block></script></block-definition><block-definition s="$robot set
position(s) 'position' of motor(s) 'motors' in
'duration' seconds | wait ? 'wait'"
type="command" category="control"><comment x="0" y="0"
w="230.00000099999988" collapsed="false">set motor to the desired
position (angle) in the desired time.&#xD;&#xD;The motor(s) input
can accept :&#xD;- string of a motor name&#xD;- string of many
motors separed with spaces&#xD;- a Snap! list of motors like the
reporter block &quot;$robot all motors&quot;</comment><header></
header><code></code><translations></translations><inputs><input
type="%n">0</input><input type="%s">motor_name</input><input
type="%n">2</input><input type="%boolUE"></input></
inputs><script><block s="doDeclareVariables"><list><l>output</
l><l>complete motor list</l><l>i</l><l>mot_pos</l><l>query_URL</
l></list></block><block s="doSetVar"><l>mot_pos</l><block
s="reportNewList"><list></list></block></block><block
s="doSetVar"><l>i</l><l>1</l></block><block s="doSetVar"><l>complete
motor list</l><custom-block s="$robot all motors"></custom-block></
block><block s="doIf"><block s="reportNot"><block
s="reportIsA"><block var="motors"/><l><option>list</option></l></
block></block><script><block s="doIfElse"><block
s="reportIsIdentical"><block var="motors"/><l></l></
block><script><block s="doSetVar"><l>motors</l><custom-block
s="$robot all motors"></custom-block></block></script><script><block
s="doSetVar"><l>motors</l><block s="reportTextSplit"><block
var="motors"/><l><option>whitespace</option></l></block></block></
script></block></script></block><custom-block s="for each %upvar of
%l %cs"><l>m</l><block var="motors"/><script><block
s="doIfElse"><block s="reportNot"><block
s="reportListContainsItem"><block var="complete motor list"/><block
var="m"/></block></block><script><custom-block s="$robot popup
%txt"><block s="reportJoinWords"><list><l>Motor </l><block var="m"/
><l> is not in the aivable motors of your robot : </l><custom-
block s="combine with %repRing items of %l"><block
s="reifyReporter"><autolambda><custom-block s="join words

```

```
%mult%txt"><list><l></l><l></l></list></custom-block></
autolambda><list></list></block><block var="complete motor list"/></
custom-block></list></block></custom-block><block s="doReport"><l></
l></block></script><script><block s="doIfElse"><block
s="reportIsA"><block var="position"/><l><option>list</option></l></
block><script><block s="doAddToList"><block
s="reportJoinWords"><list><block var="m"/><l>:</l><block
s="reportListItem"><block var="i"/><block var="position"/></block></
list></block><block var="mot_pos"/></block></script><script><block
s="doAddToList"><block s="reportJoinWords"><list><block var="m"/
><l>:</l><block var="position"/></list></block><block var="mot_pos"/
></block></script></block></script></block><block
s="doChangeVar"><l>i</l><l>1</l></block></script></custom-
block><block s="doSetVar"><l>query_URL</l><block
s="reportURL"><block s="reportJoinWords"><list><custom-block
s="$robot robot URL"></custom-block><l>/motors/set/goto/</l><block
s="reportJoinWords"><list><custom-block s="combine with %repRing
items of %l"><block s="reifyReporter"><autolambda><block
s="reportJoinWords"><list><block var="#1"/><l>:</l><block
var="duration"/><l>;</l><block var="#2"/></list></block></
autolambda><list><l>#1</l><l>#2</l></list></block><block
var="mot_pos"/></custom-block><l>:</l><block var="duration"/></
list></block></list></block></block></block><block
s="doIfElse"><block s="evaluate"><block var="wait"/><list></list></
block><script><block s="doRun"><block
s="reifyReporter"><autolambda><block var="query_URL"/></
autolambda><list></list></block><list></list></block><block
s="doWait"><block var="duration"/></block></script><script><block
s="doReport"><block var="query_URL"/></block></script></block></
script></block-definition><block-definition s="$robot reset
'simulation'" type="command" category="control"><comment
x="0" y="0" w="182" collapsed="false">Reset the simulation in V-
rep.&#xD;It is usefull if you have connection issues with V-rep.</
comment><header></header><code></code><translations></
translations><inputs><input type="%s"
readonly="true">robot<options>robot&#xD;simulation</options></
input></inputs><script><block s="doIfElse"><block
s="reportEquals"><block var="simulation"/><l>robot</l></
block><script><block s="fork"><block
s="reifyReporter"><autolambda><block s="evaluate"><block
s="reifyReporter"><autolambda><block s="reportURL"><block
s="reportJoinWords"><list><block var="host"/><l>/services.php?
python=restart</l></list></block></block></autolambda><list></
list></block><list></list></block></autolambda><list></list></
block><list></list></block><block s="fork"><block
s="reifyReporter"><autolambda><block s="evaluate"><block
s="reifyReporter"><autolambda><block s="reportURL"><block
s="reportJoinWords"><list><block var="host"/><l>/reset</l></list></
block></block></autolambda><list></list></block><list></list></
block></autolambda><list></list></block><list></list></block></
script><script><block s="doReport"><block s="reportURL"><block
s="reportJoinWords"><list><custom-block s="$robot robot URL"></
custom-block><l>/reset-simulation</l></list></block></block></
block></script></block></script></block-definition><block-definition
```



```

s="$robot all recorded moves" type="reporter"
category="sensing"><comment x="0" y="0" w="164"
collapsed="false">Return a list of all previous records filename.</
comment><header></header><code></code><translations></
translations><inputs></inputs><script><block s="doReport"><block
s="reportTextSplit"><block s="reportURL"><block
s="reportJoinWords"><list><custom-block s="$robot robot URL"></
custom-block><l>/primitive/MovePlayer</l></list></block></block><l>/
</l></block></block></script></block-definition><block-definition
s="append '&#x27;lists&#x27;";" type="reporter"
category="lists"><comment x="0" y="0" w="287" collapsed="false">Take
any number of input lists, and create a new list containing the
items of the input lists. So&#xD; APPEND [A B] [C
D]&#xD;where the [, , ,] are lists reports the list&#xD;
[A B C D]&#xD;not [[A B] [C D]].</comment><header></header><code></
code><translations></translations><inputs><input type="%mult%l"></
input></inputs><script><block s="doWarp"><script><block
s="doIf"><block s="reportEquals"><block var="lists"/><block
s="reportNewList"><list></list></block></block><script><block
s="doReport"><block s="reportNewList"><list></list></block></
block></script></block><block s="doIf"><block
s="reportEquals"><block s="reportListLength"><block var="lists"/></
block><l>1</l></block><script><block s="doReport"><block
s="reportListItem"><l>1</l><block var="lists"/></block></block></
script></block><block s="doIf"><block s="reportEquals"><block
s="reportListItem"><l>1</l><block var="lists"/></block><block
s="reportNewList"><list></list></block></block><script><block
s="doReport"><custom-block s="append '%mult%l'"><block
s="reportCDR"><block var="lists"/></block></custom-block></block></
script></block><block s="doReport"><block s="reportCONS"><block
s="reportListItem"><l>1</l><block s="reportListItem"><l>1</l><block
var="lists"/></block></block><custom-block s="append '%mult%l'"><block
s="reportCONS"><block s="reportCDR"><block s="reportListItem"><l>1</
l><block var="lists"/></block></block><block s="reportCDR"><block
var="lists"/></block></block></custom-block></block></block></
script></block></script></block-definition><block-definition
s="reverse '&#x27;data&#x27;";" type="reporter"
category="lists"><comment x="0" y="0" w="223"
collapsed="false">Reports a new list containing the items of the
input list, but in the opposite order.</comment><header></
header><code></code><translations></translations><inputs><input
type="%l"></input></inputs><script><block
s="doDeclareVariables"><list><l>helper</l></list></block><block
s="doSetVar"><l>helper</l><block s="reifyScript"><script><block
s="doIf"><block s="reportEquals"><block var="input"/><block
s="reportNewList"><list></list></block></block><script><block
s="doReport"><block var="result"/></block></script></block><block
s="doReport"><block s="evaluate"><block var="helper"/><list><block
s="reportCDR"><block var="input"/></block><block
s="reportCONS"><block s="reportListItem"><l>1</l><block var="input"/
></block><block var="result"/></block></list></block></block></
script><list><l>input</l><l>result</l></list></block></block><block
s="doWarp"><script><block s="doReport"><block s="evaluate"><block
var="helper"/><list><block var="data"/><block

```

```

s="reportNewList"><list></list></block></list></block></block></
script></block></script></block-definition><block-definition
s="remove duplicates from %&apos;data&apos;" type="reporter"
category="lists"><comment x="0" y="0" w="209"
collapsed="false">Reports a new list whose items are the same as in
the input list, except that if two or more equal items appear in the
input list, only the last one is kept in the result.</
comment><header></header><code></code><translations></
translations><inputs><input type="%l"></input></
inputs><script><block s="doWarp"><script><block s="doIf"><block
s="reportEquals"><block var="data"/><block
s="reportNewList"><list></list></block></block><script><block
s="doReport"><block s="reportNewList"><list></list></block></
block></script></block><block s="doIfElse"><block
s="reportListContainsItem"><block s="reportCDR"><block var="data"/
></block><block s="reportListItem"><l>1</l><block var="data"/></
block></block><script><block s="doReport"><custom-block s="remove
duplicates from %l"><block s="reportCDR"><block var="data"/></
block></custom-block></block></script><script><block
s="doReport"><block s="reportCONS"><block s="reportListItem"><l>1</
l><block var="data"/></block><custom-block s="remove duplicates from
%l"><block s="reportCDR"><block var="data"/></block></custom-
block></block></block></script></block></script></block></script></
block-definition><block-definition s="sort %&apos;data&apos;
ordering with %&apos;function&apos;" type="reporter"
category="lists"><comment x="0" y="0" w="204"
collapsed="false">Reports a sorted version of the list in its first
input slot, using the comparison function in the second input slot.
For a list of numbers, using &lt; as the comparison function will
sort from low to high; using &gt; will sort from high to low.</
comment><header></header><code></code><translations></
translations><inputs><input type="%l"></input><input
type="%predRing"></input></inputs><script><block
s="doDeclareVariables"><list><l>even items</l><l>odd items</
l><l>merge</l></list></block><block s="doSetVar"><l>odd items</
l><block s="reifyScript"><script><block s="doIf"><block
s="reportEquals"><l></l><block s="reportNewList"><list></list></
block></block><script><block s="doReport"><block
s="reportNewList"><list></list></block></block></script></
block><block s="doReport"><block s="reportCONS"><block
s="reportListItem"><l>1</l><l></l></block><block s="evaluate"><block
var="even items"/><list><block s="reportCDR"><l></l></block></list></
block></block></script><list></list></block></block><block
s="doSetVar"><l>even items</l><block s="reifyScript"><script><block
s="doIf"><block s="reportEquals"><l></l><block
s="reportNewList"><list></list></block></block><script><block
s="doReport"><block s="reportNewList"><list></list></block></
block></script></block><block s="doReport"><block
s="evaluate"><block var="odd items"/><list><block s="reportCDR"><l/
></block></list></block></block></script><list></list></block></
block><block s="doSetVar"><l>merge</l><block
s="reifyScript"><script><block s="doIf"><block
s="reportEquals"><block var="#1"/><block s="reportNewList"><list></
list></block></block><script><block s="doReport"><block var="#2"/></

```

```

block></script></block><block s="doIf"><block
s="reportEquals"><block var="#2"/><block s="reportNewList"><list></
list></block></block><script><block s="doReport"><block var="#1"/></
block></script></block><block s="doIfElse"><block
s="evaluate"><block var="function"/><list><block
s="reportListItem"><l>1</l><block var="#1"/></block><block
s="reportListItem"><l>1</l><block var="#2"/></block></list></
block><script><block s="doReport"><block s="reportCONS"><block
s="reportListItem"><l>1</l><block var="#1"/></block><block
s="evaluate"><block var="merge"/><list><block s="reportCDR"><block
var="#1"/></block><block var="#2"/></list></block></block></block></
script><script><block s="doReport"><block s="reportCONS"><block
s="reportListItem"><l>1</l><block var="#2"/></block><block
s="evaluate"><block var="merge"/><list><block var="#1"/><block
s="reportCDR"><block var="#2"/></block></list></block></block></
block></script></block></script><list><l>#1</l><l>#2</l></list></
block></block><block s="doIf"><block s="reportEquals"><block
var="data"/><block s="reportNewList"><list></list></block></
block><script><block s="doReport"><block s="reportNewList"><list></
list></block></block></script></block><block s="doIf"><block
s="reportEquals"><block s="reportCDR"><block var="data"/></
block><block s="reportNewList"><list></list></block></
block><script><block s="doReport"><block var="data"/></block></
script></block><block s="doReport"><block s="evaluate"><block
var="merge"/><list><custom-block s="sort %l ordering with
%predRing"><block s="evaluate"><block var="odd items"/><list><block
var="data"/></list></block><block var="function"/></custom-
block><custom-block s="sort %l ordering with %predRing"><block
s="evaluate"><block var="even items"/><list><block var="data"/></
list></block><block var="function"/></custom-block></list></block></
block></script></block-definition><block-definition s="$robot set
%&apos;register&apos; of motor(s) %&apos;motors&apos; to
%&apos;value&apos;" type="command" category="control"><comment x="0"
y="0" w="230.00000099999988" collapsed="false">Set a defined
<value> to a defined <register> of one or many
motors.&#xD;Example of some registers : torque_limit, angle_limit,
safe_compliant, compliant...&#xD;For more informations about
registers, look at the online documentation : http://$robot-
project.github.io/pypot/&#xD;&#xD;The motor(s) input can
accept :&#xD;- string of a motor name&#xD;- string of many motors
separed with spaces&#xD;- a Snap! list of motors like the reporter
block &quot;$robot all motors&quot;</comment><header></
header><code></code><translations></translations><inputs><input
type="%txt"><options>position=goal_position&#xD;speed=moving_speed&#
xD;max torque(%)=torque_limit&#xD;compliance(True or
False)=compliant</options></input><input type="%s">motor_name</
input><input type="%s">value(s)</input></inputs><script><block
s="doDeclareVariables"><list><l>output</l><l>complete motor list</
l></list></block><block s="doSetVar"><l>complete motor list</
l><custom-block s="$robot all motors"></custom-block></block><block
s="doIf"><block s="reportNot"><block s="reportIsA"><block
var="motors"/><l><option>list</option></l></block></
block><script><block s="doIfElse"><block
s="reportIsIdentical"><block var="motors"/><l>all</l></

```

```

block><script><block s="doSetVar"><l>motors</l><custom-block
s="$robot all motors"></custom-block></block></script><script><block
s="doSetVar"><l>motors</l><block s="reportTextSplit"><block
var="motors"/><l><option>whitespace</option></l></block></block></
script></block></script></block><custom-block s="for each %upvar of
%l %cs"><l>m</l><block var="motors"/><script><block s="doIf"><block
s="reportNot"><block s="reportListContainsItem"><block var="complete
motor list"/><block var="m"/></block></block><script><custom-block
s="$robot popup %txt"><block s="reportJoinWords"><list><l>Motor &lt;
</l><block var="m"/><l> &gt; is not in the aivaible motors of your
robot : </l><custom-block s="combine with %repRing items of
%l"><block s="reifyReporter"><autolambda><custom-block s="join words
%mult%txt"><list><l></l><l></l></list></custom-block></
autolambda><list></list></block><block var="complete motor list"/></
custom-block></list></block></custom-block><block s="doReport"><l></
l></block></script></block></script></custom-block><block
s="doReport"><block s="reportURL"><block
s="reportJoinWords"><list><custom-block s="$robot robot URL"></
custom-block><l>/motors/set/registers/</l><block
s="reportJoinWords"><list><custom-block s="combine with %repRing
items of %l"><block s="reifyReporter"><autolambda><block
s="reportJoinWords"><list><block var="#1"/><l>:</l><block
var="register"/><l>:</l><block var="value"/><l>;</l><block var="#2"/
></list></block></autolambda><list><l>#1</l><l>#2</l></list></
block><block var="motors"/></custom-block><l>:</l><block
var="register"/><l>:</l><block var="value"/></list></block></list></
block></block></block></script></block-definition><block-definition
s="$robot get %&apos;running&apos; behaviours" type="reporter"
category="sensing"><header></header><code></code><translations></
translations><inputs><input type="%s"
readonly="true">all<options>all&#xD;running</options></input></
inputs><script><block s="doIfElse"><block
s="reportIsIdentical"><block var="running"/><l>running</l></
block><script><block s="doReport"><block s="reportTextSplit"><block
s="reportURL"><block s="reportJoinWords"><list><custom-block
s="$robot robot URL"></custom-block><l>/primitives/</l><l>running</
l></list></block></block><l></l></block></block></
script><script><block s="doReport"><block s="reportTextSplit"><block
s="reportURL"><block s="reportJoinWords"><list><custom-block
s="$robot robot URL"></custom-block><l>/primitives</l><l></l></
list></block></block><l></l></block></block></script></block></
script></block-definition><block-definition s="$robot play move
%&apos;record_name&apos; | speed x %&apos;speed&apos;"
type="reporter" category="control"><comment x="0" y="0"
w="230.00000099999988" collapsed="false">Play the movement of a
previous recorded move. &#xD;&#xD;The &lt;record_name&gt; input is
the name of a .record filename which should be in the same directory
where you started the web server (with the command &quot;$robot-snap
$robot-ergo&quot; for example).&#xD;&#xD;You can change the play
speed with the &lt;speed&gt; input.</comment><header></
header><code></code><translations></translations><inputs><input
type="%txt">move_name</input><input type="%n">1</input></
inputs><script><block s="doDeclareVariables"><list><l>time</l></
list></block><block s="doDeclareVariables"><list><l>wait</l></

```

```

list></block><block s="doSetVar"><l>wait</l><block
s="reportBoolean"><l><bool>>true</bool></l></block></block><block
s="doIfElse"><block s="reportListContainsItem"><custom-block
s="$robot all recorded moves"></custom-block><block
var="record_name"/></block><script><block s="doSetVar"><l>time</
l><block s="reportURL"><block s="reportJoinWords"><list><custom-
block s="$robot robot URL"></custom-block><l>/primitive/MovePlayer/
</l><block var="record_name"/><l>/start</l><block var="speed"/></
list></block></block></block><block s="doIf"><block var="wait"/
><script><block s="doWait"><block var="time"/></block></
script><comment w="90" collapsed="false">ugly wait to stop
feature.&#xD;TODO : change with a looking at running primitive</
comment></block></script><script><custom-block s="$robot popup
%txt"><block s="reportJoinWords"><list><l>Error : &lt;</l><block
var="record_name"/><l>&gt; is not is the avaible records (look at
the block $robot all records to see all recorded moves) </l><custom-
block s="$robot all recorded moves"></custom-block></list></block></
custom-block></script></block><block s="doReport"><block
var="record_name"/></block></script></block-definition><block-
definition s="$robot sequence %&apos;mov&apos;" type="reporter"
category="control"><header></header><code></code><translations></
translations><inputs><input type="%mult%repRing"></input></
inputs><script><block s="doDeclareVariables"><list><l>out</l></
list></block><block s="doSetVar"><l>out</l><l></l></block><custom-
block s="for each %upvar of %l %cs"><l>item</l><block var="mov"/
><script><block s="doSetVar"><l>out</l><block
s="reportJoinWords"><list><block s="reportCallCC"><block var="item"/
></block><l></l><block var="out"/></list></block></block></
script></custom-block><block s="doReport"><block var="out"/></
block></script></block-definition><block-definition s="$robot
concurrent %&apos;movements&apos;" type="reporter"
category="control"><comment x="0" y="0" w="201.99999999999997"
collapsed="false">All reports input are run concurently.&#xD;You can
use this block to play concurently many recorded move</
comment><header></header><code></code><translations></
translations><inputs><input type="%mult%repRing"></input></
inputs><script><block s="doDeclareVariables"><list><l>results</
l><l>iterator</l></list></block><block s="doSetVar"><l>results</
l><block s="reportNewList"><list></list></block></block><block
s="doSetVar"><l>iterator</l><l>0</l></block><block
s="doRepeat"><block s="reportListLength"><block var="movements"/></
block><script><block s="fork"><block s="reifyScript"><script><block
s="doAddToList"><block s="reportCallCC"><block
s="reportListItem"><block var="iterator"/><block var="movements"/></
block></block><block var="results"/></block></script><list></list></
block><list></list></block><block s="doChangeVar"><l>iterator</
l><l>1</l></block><custom-block s="yield"></custom-block></script></
block><block s="doWaitUntil"><block s="reportEquals"><block
s="reportListLength"><block var="movements"/></block><block
s="reportListLength"><block var="results"/></block></block></
block><block s="doReport"><block var="results"/></block></script></
block-definition><block-definition s="map %&apos;function&apos; over
%&apos;lists&apos;" type="reporter" category="lists"><header></
header><code></code><translations></translations><inputs><input

```

```
type="%repRing"></input><input type="%mult%l"></input></
inputs><script><block s="doWarp"><script><block
s="doDeclareVariables"><list><l>mapone</l><l>mapmany</l></list></
block><block s="doSetVar"><l>mapone</l><block
s="reifyScript"><script><block s="doIf"><custom-block s="empty?
%l"><block var="data"/></custom-block><script><block
s="doReport"><block s="reportNewList"><list></list></block></
block></script></block><block s="doReport"><block
s="reportCONS"><block s="evaluate"><block var="function"/
><list><block s="reportListItem"><l>1</l><block var="data"/></
block></list></block><block s="evaluate"><block var="mapone"/
><list><block s="reportCDR"><block var="data"/></block></list></
block></block></block></script><list><l>data</l></list></block></
block><block s="doSetVar"><l>mapmany</l><block
s="reifyScript"><script><block s="doIf"><custom-block s="empty?
%l"><block s="reportListItem"><l>1</l><block var="data lists"/></
block></custom-block><script><block s="doReport"><block
s="reportNewList"><list></list></block></block></script></
block><block s="doReport"><block s="reportCONS"><block
s="evaluate"><block var="function"/><custom-block s="map %repRing
over %mult%l"><block s="reifyReporter"><autolambda><block
s="reportListItem"><l>1</l><l></l></block></autolambda><list></list></
block><list><block var="data lists"/></list></custom-block></
block><block s="evaluate"><block var="mapmany"/><list><custom-block
s="map %repRing over %mult%l"><block
s="reifyReporter"><autolambda><block s="reportCDR"><l></l></block></
autolambda><list></list></block><list><block var="data lists"/></
list></custom-block></list></block></block></block></
script><list><l>data lists</l></list></block></block><block
s="doIfElse"><custom-block s="empty? %l"><block s="reportCDR"><block
var="lists"/></block></custom-block><script><block
s="doReport"><block s="evaluate"><block var="mapone"/><list><block
s="reportListItem"><l>1</l><block var="lists"/></block></list></
block></block></script><script><block s="doReport"><block
s="evaluate"><block var="mapmany"/><list><block var="lists"/></
list></block></block></script></block></script></block></script></
block-definition><block-definition s="empty? %&apos;data&apos;"
type="predicate" category="lists"><header></header><code></
code><translations></translations><inputs><input type="%l"></
input></inputs><script><block s="doReport"><block
s="reportEquals"><block var="data"/><block
s="reportNewList"><list></list></block></block></block></script></
block-definition><block-definition s="keep items such that
%&apos;pred&apos; from %&apos;data&apos;" type="reporter"
category="lists"><header></header><code></code><translations></
translations><inputs><input type="%predRing"></input><input
type="%l"></input></inputs><script><block s="doWarp"><script><block
s="doIf"><custom-block s="empty? %l"><block var="data"/></custom-
block><script><block s="doReport"><block s="reportNewList"><list></
list></block></block></script></block><block s="doIfElse"><block
s="evaluate"><block var="pred"/><list><block
s="reportListItem"><l>1</l><block var="data"/></block></list></
block><script><block s="doReport"><block s="reportCONS"><block
s="reportListItem"><l>1</l><block var="data"/></block><custom-block
```

```
s="keep items such that %predRing from %l"><block var="pred"/><block
s="reportCDR"><block var="data"/></block></custom-block></block></
block></script><script><block s="doReport"><custom-block s="keep
items such that %predRing from %l"><block var="pred"/><block
s="reportCDR"><block var="data"/></block></custom-block></block></
script></block></script></block></script></block-definition><block-
definition s="combine with %&apos;function&apos; items of
%&apos;data&apos;" type="reporter" category="lists"><header></
header><code></code><translations></translations><inputs><input
type="%repRing"></input><input type="%l"></input></
inputs><script><block s="doWarp"><script><block s="doIf"><custom-
block s="empty? %l"><block s="reportCDR"><block var="data"/></
block></custom-block><script><block s="doReport"><block
s="reportListItem"><l>1</l><block var="data"/></block></block></
script></block><block s="doReport"><block s="evaluate"><block
var="function"/><list><block s="reportListItem"><l>1</l><block
var="data"/></block><custom-block s="combine with %repRing items of
%l"><block var="function"/><block s="reportCDR"><block var="data"/
></block></custom-block></list></block></block></script></block></
script></block-definition><block-definition s="join words
%&apos;words&apos;" type="reporter" category="operators"><header></
header><code></code><translations></translations><inputs><input
type="%mult%txt"></input></inputs><script><block
s="doWarp"><script><block s="doIf"><custom-block s="empty?
%l"><block s="reportCDR"><block var="words"/></block></custom-
block><script><block s="doReport"><block s="reportListItem"><l>1</
l><block var="words"/></block></block></script></block><block
s="doReport"><block s="reportJoinWords"><list><block
s="reportListItem"><l>1</l><block var="words"/></block><block
s="reportJoinWords"><list><l> </l><block s="evaluate"><block
s="reifyReporter"><autolambda><custom-block s="join words
%mult%txt"><block s="reportCDR"><block var="words"/></block></
custom-block></autolambda><list></list></block><list></list></
block></list></block></list></block></block></script></block></
script></block-definition><block-definition s="list $arrowRight
sentence %&apos;data&apos;" type="reporter"
category="operators"><header></header><code></code><translations></
translations><inputs><input type="%l"></input></
inputs><script><block s="doWarp"><script><block
s="doReport"><custom-block s="combine with %repRing items of
%l"><block s="reifyReporter"><autolambda><custom-block s="join words
%mult%txt"><list><l></l><l></l></list></custom-block></
autolambda><list></list></block><block var="data"/></custom-block></
block></script></block></script></block-definition><block-definition
s="sentence $arrowRight list %&apos;text&apos;" type="reporter"
category="operators"><header></header><code></code><translations></
translations><inputs><input type="%txt"></input></
inputs><script><block s="doWarp"><script><block s="doReport"><block
s="reportTextSplit"><block var="text"/><l><option>whitespace</
option></l></block></block></script></block></script></block-
definition><block-definition s="for each %&apos;item&apos; of
%&apos;data&apos; %&apos;action&apos;" type="command"
category="lists"><header></header><code></code><translations></
translations><inputs><input type="%upvar"></input><input
```

```

type="%l"></input><input type="%cs"></input></inputs><script><block
s="doUntil"><custom-block s="empty? %l"><block var="data"/></custom-
block><script><block s="doSetVar"><l>item</l><block
s="reportListItem"><l>1</l><block var="data"/></block></block><block
s="doRun"><block var="action"/><list><block
s="reportListItem"><l>1</l><block var="data"/></block></list></
block><block s="doSetVar"><l>data</l><block s="reportCDR"><block
var="data"/></block></block></script></block></script></block-
definition><block-definition s="word $arrowRight list
%&apos;word&apos;" type="reporter" category="operators"><header></
header><code></code><translations></translations><inputs><input
type="%txt"></input></inputs><script><block
s="doWarp"><script><block s="doReport"><block
s="reportTextSplit"><block var="word"/><l><option>letter</option></
l></block></block></script></block></script></block-
definition><block-definition s="sum %&apos;nums&apos;"
type="reporter" category="operators"><comment x="0" y="0"
w="201.99999999999997" collapsed="false">Takes any number of numbers
as inputs (use the left and right arrowheads to adjust the number of
input slots) and reports the result of adding them all, so&#xD;ISUM
(4) (100) (8))&#xD;reports 112.</comment><header></header><code></
code><translations></translations><inputs><input type="%mult%n"></
input></inputs><script><block s="doIf"><block
s="reportEquals"><block var="nums"/><block
s="reportNewList"><list></list></block></block><script><block
s="doReport"><l>0</l></block></script></block><block
s="doReport"><block s="reportSum"><block s="reportListItem"><l>1</
l><block var="nums"/></block><custom-block s="sum %mult%n"><block
s="reportCDR"><block var="nums"/></block></custom-block></block></
block></script></block-definition><block-definition s="product
%&apos;nums&apos;" type="reporter" category="operators"><comment
x="0" y="0" w="204.99999999999997" collapsed="false">Takes any
number of numbers as inputs (use the left and right arrowheads to
adjust the number of input slots) and reports the result of
multiplying them all, so&#xD;(PRODUCT (4) (100) (8))&#xD;reports
3200.</comment><header></header><code></code><translations></
translations><inputs><input type="%mult%n"></input></
inputs><script><block s="doIf"><block s="reportEquals"><block
var="nums"/><block s="reportNewList"><list></list></block></
block><script><block s="doReport"><l>1</l></block></script></
block><block s="doReport"><block s="reportProduct"><block
s="reportListItem"><l>1</l><block var="nums"/></block><custom-block
s="product %mult%n"><block s="reportCDR"><block var="nums"/></
block></custom-block></block></block></script></block-
definition><block-definition s="all of %&apos;preds&apos;"
type="predicate" category="operators"><comment x="0" y="0"
w="247.99999999999997" collapsed="false">Takes any number of Boolean
(true/false) inputs (use the left and right arrowheads to adjust the
number of input slots) and reports TRUE only if all of the inputs
are TRUE, otherwise FALSE. Like AND but for multiple inputs.</
comment><header></header><code></code><translations></
translations><inputs><input type="%mult%boolUE"></input></
inputs><script><block s="doIf"><block s="reportEquals"><block
var="preds"/><block s="reportNewList"><list></list></block></

```



```

block><script><block s="doReport"><block
s="reportBoolean"><l><bool>>true</bool></l></block></block></
script></block><block s="doReport"><block s="reportAnd"><block
s="evaluate"><block s="reportListItem"><l>1</l><block var="preds"/
></block><list></list></block><custom-block s="all of
%mult%boolUE"><block s="reportCDR"><block var="preds"/></block></
custom-block></block></block></script></block-definition><block-
definition s="any of %&apos;preds&apos;" type="predicate"
category="operators"><comment x="0" y="0" w="240"
collapsed="false">Takes any number of Boolean (true/false) inputs
(use the left and right arrowheads to adjust the number of input
slots) and reports TRUE if at least one input is TRUE, otherwise
FALSE.</comment><header></header><code></code><translations></
translations><inputs><input type="%mult%boolUE"></input></
inputs><script><block s="doIf"><block s="reportEquals"><block
var="preds"/><block s="reportNewList"><list></list></block></
block><script><block s="doReport"><block
s="reportBoolean"><l><bool>>false</bool></l></block></block></
script></block><block s="doReport"><block s="reportOr"><block
s="evaluate"><block s="reportListItem"><l>1</l><block var="preds"/
></block><list></list></block><custom-block s="any of
%mult%boolUE"><block s="reportCDR"><block var="preds"/></block></
custom-block></block></block></script></block-definition><block-
definition s="yield" type="command" category="other"><header></
header><code></code><translations></translations><inputs></
inputs><script><block s="doWait"><l>0.00000000000001</l></block></
script></block-definition><block-definition s="$robot create &
start record move %&apos;record_name&apos; with motor(s)
%&apos;motors input&apos;" type="command"
category="control"><comment x="0" y="0" w="230.00000099999988"
collapsed="false">Create and start the record of a definied
move..&#xD;&#xD;The &lt;record_name&gt; input is used to store the
recorded move. The move is stored in the directory where you started
the web server (with the command &quot;$robot-snap $robot-ergo&quot;
for example), and can be replayed even if you close Snap! &#xD;
&#xD;The motor(s) input can accept :&#xD;- string of a motor
name&#xD;- string of many motors separed with spaces&#xD;- a Snap!
list of motors like the reporter block &quot;$robot all
motors&quot;&#xD;&#xD;If the motor(s) input is empty, all motors
will be recorded.</comment><header></header><code></
code><translations></translations><inputs><input
type="%s">move_name</input><input type="%s"></input></
inputs><script><block s="doDeclareVariables"><list><l>motors_list</
l><l>complete motor list</l><l>complete aliases list</l></list></
block><block s="doSetVar"><l>motors_list</l><block
s="reportNewList"><list></list></block></block><block
s="doSetVar"><l>complete motor list</l><custom-block s="$robot all
motors"></custom-block></block><block s="doIf"><block
s="reportIsIdentical"><block var="record_name"/><l></l></
block><script><custom-block s="$robot popup %txt"><l>Input error,
record name must not be empty</l></custom-block><block
s="doReport"><l></l></block></script></block><block
s="doIfElse"><block s="reportEquals"><block var="motors input"/
><l></l></block><script><block s="doReport"><block

```

```
s="reportURL"><block s="reportJoinWords"><list><custom-block
s="$robot robot URL"></custom-block><l>/primitive/MoveRecorder/</
l><block var="record_name"/><l>/start/</l><custom-block s="combine
with %repRing items of %l"><block
s="reifyReporter"><autolambda><block
s="reportJoinWords"><list><block var="#1"/><l>;</l><block var="#2"/
></list></block></autolambda><list><l>#1</l><l>#2</l></list></
block><custom-block s="$robot all motors"></custom-block></custom-
block></list></block></block></block></script><script><block
s="doIfElse"><block s="reportIsA"><block var="motors input"/
><l><option>list</option></l></block><script><block
s="doSetVar"><l>motors_list</l><custom-block s="append
%mult%l"><list><block var="motors_list"/><block var="motors input"/
></list></custom-block></block></script><script><block
s="doIfElse"><block s="reportOr"><block s="reportIsA"><block
var="motors input"/><l><option>number</option></l></block><block
s="reportIsA"><block var="motors input"/><l><option>text</option></
l></block></block><script><block s="doSetVar"><l>motors_list</
l><custom-block s="append %mult%l"><list><block var="motors_list"/
><block s="reportTextSplit"><block var="motors input"/
><l><option>whitespace</option></l></block></list></custom-block></
block></script><script><custom-block s="$robot popup %txt"><l>Input
error, you should look at help (right click on the bock)</l></
custom-block></script></block></script></block></script></
block><custom-block s="for each %upvar of %l %cs"><l>m</l><block
var="motors_list"/><script><block s="doIf"><block
s="reportNot"><block s="reportListContainsItem"><block var="complete
motor list"/><block var="m"/></block></block><script><block
s="doIf"><block s="reportEquals"><block var="complete aliases list"/
><l>0</l><comment w="90" collapsed="false">Set all aliases here for
optimisation issues</comment></block><script><block
s="doSetVar"><l>complete aliases list</l><custom-block s="$robot all
motors groups"></custom-block></block></script></block><block
s="doIfElse"><block s="reportListContainsItem"><block var="complete
aliases list"/><block var="m"/></block><script><block
s="doSetVar"><l>motors_list</l><custom-block s="append
%mult%l"><list><block var="motors_list"/><custom-block s="$robot
motors in group %txt"><block var="m"/></custom-block></list></
custom-block></block></script><script><custom-block s="$robot popup
%txt"><block s="reportJoinWords"><list><l>Motor </l><block var="m"/
><l>does not exist in your robot. </l></list></block></custom-
block><block s="doReport"><l></l></block></script></block></
script></block></script></custom-block><custom-block s="$robot set
motor(s) %s %txt"><block var="motors_list"/><l>compliant</l></
custom-block><block s="doReport"><block s="reportURL"><block
s="reportJoinWords"><list><custom-block s="$robot robot URL"></
custom-block><l>/primitive/MoveRecorder/</l><block
var="record_name"/><l>/start/</l><custom-block s="combine with
%repRing items of %l"><block s="reifyReporter"><autolambda><block
s="reportJoinWords"><list><block var="#1"/><l>;</l><block var="#2"/
></list></block></autolambda><list><l>#1</l><l>#2</l></list></
block><block var="motors_list"/></custom-block></list></block></
block></block></script></block-definition><block-definition
s="$robot stop record & save move %&apos;record_name&apos;";"
```

```

type="command" category="control"><comment x="0" y="0"
w="489.00000000000006" collapsed="false">Stop recording the desired
move record. &#xD;Be careful, you must have previously defined a
move record with the &quot;create move ... with motor(s) ...&quot;
block</comment><header></header><code></code><translations></
translations><inputs><input type="%s">move_name</input></
inputs><script><block s="doDeclareVariables"><list><l>motors</l></
list></block><block s="doSetVar"><l>motors</l><block
s="reportTextSplit"><block s="reportURL"><block
s="reportJoinWords"><list><custom-block s="$robot robot URL"></
custom-block><l>/primitive/MoveRecorder/</l><block
var="record_name"/><l>/get_motors</l></list></block></block><l></
l></block></block><custom-block s="$robot set motor(s) %s
%txt"><block var="motors"/><block s="reportBoolean"><l><bool>>false</
bool></l></block></custom-block><block s="doReport"><block
s="reportURL"><block s="reportJoinWords"><list><custom-block
s="$robot robot URL"></custom-block><l>/primitive/MoveRecorder/</
l><block var="record_name"/><l>/stop</l></list></block></block></
block></script></block-definition><block-definition s="$robot stop
move %&apos;record_name&apos;" type="command"
category="control"><header></header><code></code><translations></
translations><inputs><input type="%s">move_name</input></
inputs><script><block s="doReport"><block s="reportURL"><block
s="reportJoinWords"><list><custom-block s="$robot robot URL"></
custom-block><l>/primitive/MovePlayer/</l><block var="record_name"/
><l>/stop</l></list></block></block></block></script></block-
definition><block-definition s="$robot popup %&apos;message&apos;"
type="command" category="other"><header></header><code></
code><translations></translations><inputs><input type="%txt"></
input></inputs><script><block s="doRun"><block
s="reportJSFunction"><list><l>message</l></list><l>alert(message);</
l></block><list><block var="message"/></list></block></script></
block-definition><block-definition s="pool map %&apos;function&apos;
over %&apos;lists&apos;" type="reporter" category="lists"><comment
x="0" y="0" w="204" collapsed="false">Works as the map block (see
map help), but it proceeds iterations concurently.</
comment><header></header><code></code><translations></
translations><inputs><input type="%repRing"></input><input
type="%mult%l"></input></inputs><script><block
s="doDeclareVariables"><list><l>results</l><l>iterator</
l><l>input_list</l><l>list</l></list></block><block
s="doSetVar"><l>results</l><block s="reportNewList"><list></list></
block></block><block s="doSetVar"><l>iterator</l><l>0</l></
block><block s="doRepeat"><block s="reportListLength"><block
s="reportListItem"><l>1</l><block var="lists"/></block></
block><script><block s="doChangeVar"><l>iterator</l><l>1</l></
block><block s="doSetVar"><l>input_list</l><block
s="reportNewList"><list></list></block></block><custom-block s="for
each %upvar of %l %cs"><l>list</l><block var="lists"/><script><block
s="doAddToList"><block s="reportListItem"><block var="iterator"/
><block var="list"/></block><block var="input_list"/></block></
script></custom-block><block s="fork"><block
s="reifyScript"><script><block s="doAddToList"><block
s="evaluate"><block var="function"/><block var="input_list"/></

```

```

block><block var="results"/></block></script><list></list></
block><list></list></block><custom-block s="yield"></custom-block></
script></block><block s="doWaitUntil"><block s="reportEquals"><block
s="reportListLength"><block var="results"/></block><block
s="reportListLength"><block s="reportListItem"><l>1</l><block
var="lists"/></block></block></block></block><block
s="doReport"><block var="results"/></block></script></block-
definition><block-definition s="$robot play concurrently moves
'inputs'" type="reporter" category="control"><comment
x="0" y="0" w="260" collapsed="false">Play previous recorded move
concurrently. Inputs can be : &#xD;* move name (string)&#xD;* many
moves name separated by whitespace&#xD;* any reporter block, like
sequential or play sequentially</comment><header></header><code></
code><translations></translations><inputs><input type="%mult%s"></
input></inputs><script><block
s="doDeclareVariables"><list><l>move_names</l><l>reporters</
l><l>tmp_string</l><l>tmp_reporter</l></list></block><block
s="doSetVar"><l>move_names</l><block s="reportNewList"><list></
list></block></block><block s="doSetVar"><l>tmp_string</l><block
s="reportNewList"><list></list></block></block><block
s="doSetVar"><l>reporters</l><block s="reportNewList"><list></
list></block></block><custom-block s="for each %upvar of %l
%cs"><l>item</l><block var="inputs"/><script><block
s="doIfElse"><block s="report0"><block s="reportIsA"><block
var="item"/><l><option>text</option></l></block><block
s="reportIsA"><block var="item"/><l><option>number</option></l></
block></block><script><block s="doSetVar"><l>move_names</l><custom-
block s="append %mult%l"><list><block var="move_names"/><block
s="reportTextSplit"><block var="item"/><l>;</l></block></list></
custom-block></block></script><script><block s="doIfElse"><block
s="reportIsA"><block var="item"/><l><option>list</option></l></
block><script><block s="doSetVar"><l>move_names</l><custom-block
s="append %mult%l"><list><block var="move_names"/><block var="item"/
></list></custom-block></block></script><script><block
s="doIfElse"><block s="reportIsA"><block var="item"/
><l><option>reporter</option></l></block><script><block
s="doAddToList"><block var="item"/><block var="reporters"/></
block></script><script><custom-block s="$robot popup %txt"><l>Input
is not valid. See help (right click on the block)</l></custom-
block></script></block></script></block></script></block></script></
custom-block><custom-block s="for each %upvar of %l %cs"><l>item</
l><block var="move_names"/><script><block s="fork"><block
s="reifyScript"><script><block s="doAddToList"><block
s="evaluate"><block s="reifyReporter"><autolambda><custom-block
s="$robot play move %txt | speed x %n"><l></l><l>1</l></custom-
block></autolambda><list></list></block><list><block var="item"/></
list></block><block var="tmp_string"/></block></script><list></
list></block><list></list></block><custom-block s="yield"></custom-
block></script></custom-block><block s="doSetVar"><l>tmp_reporter</
l><custom-block s="$robot concurrent %mult%repRing"><list><block
s="reifyReporter"><script></script><list></list></block></list></
custom-block></block><block s="doWaitUntil"><block
s="reportEquals"><block s="reportListLength"><block
var="tmp_string"/></block><block s="reportListLength"><block

```

```

var="move_names"/></block></block></block><block
s="doReport"><custom-block s="append %mult%l"><list><block
var="tmp_reporter"/><block var="tmp_string"/></list></custom-
block></block></script><scripts><comment x="131" y="660.2" w="265"
collapsed="false">TODO : understand why a sequential input blocks
return items on a list rather than separated with whitespace</
comment></scripts></block-definition><block-definition s="$robot
play sequentially moves %&apos;inputs&apos;" type="reporter"
category="control"><comment x="0" y="0" w="267"
collapsed="false">Play previous recorded moves sequentially. Inputs
can be : &#xD;* move name (string)&#xD;* many moves name separated
by whitespace&#xD;* any reporter block like concurrent or play
concurrently</comment><header></header><code></code><translations></
translations><inputs><input type="%mult%s"></input></
inputs><script><block s="doDeclareVariables"><list><l>move_names</
l><l>reporters</l><l>out</l></list></block><block
s="doSetVar"><l>move_names</l><block s="reportNewList"><list></
list></block></block><block s="doSetVar"><l>reporters</l><block
s="reportNewList"><list></list></block></block><custom-block s="for
each %upvar of %l %cs"><l>item</l><block var="inputs"/
><script><block s="doIfElse"><block s="report0r"><block
s="reportIsA"><block var="item"/><l><option>text</option></l></
block><block s="reportIsA"><block var="item"/><l><option>number</
option></l></block></block><script><block
s="doSetVar"><l>move_names</l><custom-block s="append
%mult%l"><list><block var="move_names"/><block
s="reportTextSplit"><block var="item"/><l>;</l></block></list></
custom-block></block></script><script><block s="doIfElse"><block
s="reportIsA"><block var="item"/><l><option>list</option></l></
block><script><block s="doSetVar"><l>move_names</l><custom-block
s="append %mult%l"><list><block var="move_names"/><block var="item"/
></list></custom-block></block></script><script><block
s="doIfElse"><block s="reportIsA"><block var="item"/
><l><option>reporter</option></l></block><script><block
s="doAddToList"><block var="item"/><block var="reporters"/></
block></script><script><custom-block s="$robot popup %txt"><l>Input
is not valid. See help (right click on the block)</l></custom-
block></script></block></script></block></script></block></script></
custom-block><block s="doReport"><custom-block s="join words
%mult%txt"><list><custom-block s="combine with %repRing items of
%l"><block s="reifyReporter"><autolambda><custom-block s="join words
%mult%txt"><list><l></l><l></l></list></custom-block></
autolambda><list></list></block><custom-block s="map %repRing over
%mult%l"><block s="reifyReporter"><autolambda><custom-block
s="$robot play move %txt | speed x %n"><l></l><l>1</l></custom-
block></autolambda><list></list></block><list><block
var="move_names"/></list></custom-block></custom-block><custom-block
s="$robot sequence %mult%repRing"><list><block
s="reifyReporter"><script></script><list></list></block></list></
custom-block></list></custom-block></block></script></block-
definition><block-definition s="$robot all motors groups"
type="reporter" category="sensing"><comment x="0" y="0" w="171"
collapsed="false">Return a list of all aliases.&#xD;Alias is a
keyword for representing many motors at once. </comment><header></

```

```

header><code></code><translations></translations><inputs></
inputs><script><block s="doReport"><block s="reportTextSplit"><block
s="reportURL"><block s="reportJoinWords"><list><custom-block
s="$robot robot URL"></custom-block><l>/motors/</l><l>alias</l></
list></block></block><l></l></block></block></script></block-
definition><block-definition s="items %&apos;i&apos; to
%&apos;j&apos; of %&apos;list&apos;" type="reporter"
category="lists"><comment x="0" y="0" w="146"
collapsed="false">Return a sublist which contains items in the
interval of 2 input numbers.&#xD;Original list is not modified.</
comment><header></header><code></code><translations></
translations><inputs><input type="%n"></input><input type="%n"></
input><input type="%l"></input></inputs><script><block
s="doDeclareVariables"><list><l>result</l></list></block><block
s="doSetVar"><l>result</l><block s="reportNewList"><list></list></
block></block><block s="doIf"><block s="reportLessThan"><block
var="i"/><l>0</l></block><script><block s="doSetVar"><l>i</l><block
s="reportSum"><block s="reportListLength"><block var="list"/></
block><block var="i"/></block></block></script></block><block
s="doIf"><block s="reportLessThan"><block var="j"/><l>0</l></
block><script><block s="doSetVar"><l>j</l><block
s="reportSum"><block s="reportListLength"><block var="list"/></
block><block var="j"/></block></block></script></block><custom-block
s="for %upvar = %n to %n %cs"><l>elem</l><custom-block s="if %b then
%anyUE else %anyUE"><block var="i"/><block var="i"/><l>1</l></
custom-block><custom-block s="if %b then %anyUE else %anyUE"><block
var="j"/><block var="j"/><block s="reportListLength"><block
var="list"/></block></custom-block><script><block
s="doAddToList"><block s="reportListItem"><block var="elem"/><block
var="list"/></block><block var="result"/></block></script></custom-
block><block s="doReport"><block var="result"/></block></script></
block-definition><block-definition s="$robot play move
%&apos;record_name&apos; in reverse | speed x %&apos;speed&apos;"
type="reporter" category="control"><comment x="0" y="0"
w="230.00000099999988" collapsed="false">Play in reverse the
movement of a previous recorded move. &#xD;&#xD;The
&lt;record_name&gt; input is the name of a .record filename which
should be in the same directory where you started the web server
(with the command &quot;$robot-snap $robot-ergo&quot; for
example).&#xD;&#xD;You can change the play speed with the
&lt;speed&gt; input.</comment><header></header><code></
code><translations></translations><inputs><input
type="%s">move_name</input><input type="%n">1</input></
inputs><script><block s="doDeclareVariables"><list><l>time</l></
list></block><block s="doDeclareVariables"><list><l>wait</l></
list></block><block s="doSetVar"><l>wait</l><block
s="reportBoolean"><l><bool>>true</bool></l></block></block><block
s="doIfElse"><block s="reportListContainsItem"><custom-block
s="$robot all recorded moves"></custom-block><block
var="record_name"/></block><script><block s="doSetVar"><l>time</
l><block s="reportURL"><block s="reportJoinWords"><list><custom-
block s="$robot robot URL"></custom-block><l>/primitive/MovePlayer/
</l><block var="record_name"/><l>/start/</l><block var="speed"/><l>
backwards</l></list></block></block></block><block s="doIf"><block

```

```

var="wait"/><script><block s="doWait"><block var="time"/></block></
script><comment w="90" collapsed="false">ugly wait to stop
feature.&#xD;TODO : change with a looking at running primitive</
comment></block></script><script><custom-block s="$robot popup
%txt"><block s="reportJoinWords"><list><l>Error : &lt;/l><block
var="record_name"/><l>&gt; is not is the avaible records (look at
the block $robot all records to see all recorded moves) </l><custom-
block s="$robot all recorded moves"></custom-block></list></block></
custom-block></script></block><block s="doReport"><block
var="record_name"/></block></script></block-definition><block-
definition s="$robot test connection" type="reporter"
category="sensing"><header></header><code></code><translations></
translations><inputs></inputs><script><block s="doIfElse"><custom-
block s="ping http:// %s"><custom-block s="$robot robot URL"></
custom-block></custom-block><script><block
s="doReport"><l>Connection ok!</l></block></script><script><block
s="doReport"><l>You may have connection troubles. Check the host
variable</l></block></script></block></script></block-
definition><block-definition s="index of %&apos;var&apos; in
%&apos;list&apos;" type="reporter" category="lists"><comment x="0"
y="0" w="194" collapsed="false">Return the index of a item in a
list</comment><header></header><code></code><translations></
translations><inputs><input type="%s"></input><input type="%l"></
input></inputs><script><block s="doDeclareVariables"><list><l>i</
l></list></block><block s="doSetVar"><l>i</l><l>1</l></
block><custom-block s="for each %upvar of %l %cs"><l>m</l><block
var="list"/><script><block s="doIf"><block s="reportEquals"><block
var="m"/><block var="var"/></block><script><block
s="doReport"><block var="i"/></block></script></block><block
s="doChangeVar"><l>i</l><l>1</l></block></script></custom-
block><block s="doReport"><l>0</l></block></script></block-
definition><block-definition s="$robot http:// %&apos;input&apos;"
type="reporter" category="other"><header></header><code></
code><translations></translations><inputs><input type="%s"></
input></inputs><script><block
s="doDeclareVariables"><list><l>output</l><l>timer_init</l></list></
block><block s="doSetVar"><l>timer_init</l><block s="getTimer"></
block></block><block s="doSetVar"><l>output</l><block
s="reportNewList"><list></list></block></block><block
s="fork"><block s="reifyScript"><script><block
s="doAddToList"><block s="reportURL"><block var="input"/></
block><block var="output"/></block></script><list></list></
block><list></list></block><block s="doWaitUntil"><block
s="report0r"><block s="reportGreaterThan"><block
s="reportDifference"><block s="getTimer"></block><block
var="timer_init"/></block><l>3.5</l></block><block
s="reportGreaterThan"><block s="reportListLength"><block
var="output"/></block><l>0</l></block></block></block><block
s="doIf"><block s="reportGreaterThan"><block
s="reportDifference"><block s="getTimer"></block><block
var="timer_init"/></block><l>3</l></block><script><custom-block
s="$robot popup %txt"><block s="reportJoinWords"><list><l>Robot is
not connected to </l><block var="input"/></list></block></custom-
block><block s="doReport"><l>error</l></block></script></

```

```
block><block s="doReport"><block s="reportListItem"><l>1</l><block
var="output"/></block></script><scripts><script
x="452.98535356250017" y="75.0000070000001"><custom-block s="$robot
all motors"></custom-block></script><script x="539.8916025624999"
y="95.00000100000011"><block s="reportURL"><block var="input"/></
block></script><script x="397.2265645" y="240.00000399999993"><block
s="doSetVar"><l>output</l><l>0</l></block></script><script x="433"
y="179.2"><block s="reportListLength"><l/></block></script><script
x="701.5517588124999" y="139.00000500000004"><block
s="reportStringSize"><l>world</l></block></script></scripts></block-
definition><block-definition s="$robot %&apos;action&apos;
behaviours %&apos;primitive&apos;" type="command"
category="control"><header></header><code></code><translations></
translations><inputs><input type="%txt"
readonly="true">start<options>start&#xD;stop&#xD;pause&#xD;resume</
options></input><input type="%txt"></input></inputs><script><block
s="doDeclareVariables"><list><l>all primitives</l><l>keyword list</
l></list></block><block s="doSetVar"><l>all primitives</l><custom-
block s="$robot get %s behaviours"><l>all</l></custom-block></
block><block s="doSetVar"><l>keyword list</l><block
s="reportNewList"><list><l>start</l><l>stop</l><l>pause</
l><l>resume</l></list></block></block><block s="doIfElse"><block
s="reportNot"><block s="reportListContainsItem"><block var="all
primitives"/><block var="primitive"/></block></
block><script><custom-block s="$robot popup %txt"><block
s="reportJoinWords"><list><l>Primitive &lt; </l><block
var="primitive"/><l> &gt; is not in the aivable primitives of your
robot : </l><custom-block s="combine with %repRing items of
%l"><block s="reifyReporter"><autolambda><custom-block s="join words
%mult%txt"><list><l></l><l></l></list></custom-block></
autolambda><list></list></block><block var="all primitives"/></
custom-block></list></block></custom-block><block s="doReport"><l></
l></block></script><script><block s="doIfElse"><block
s="reportNot"><block s="reportListContainsItem"><block var="keyword
list"/><block var="action"/></block></block><script><custom-block
s="$robot popup %txt"><block s="reportJoinWords"><list><l>Action
&lt; </l><block var="action"/><l> &gt; is not valid : </l><custom-
block s="combine with %repRing items of %l"><block
s="reifyReporter"><autolambda><custom-block s="join words
%mult%txt"><list><l></l><l></l></list></custom-block></
autolambda><list></list></block><block var="keyword list"/></custom-
block></list></block></custom-block><block s="doReport"><l></l></
block></script><script><block s="doReport"><block
s="reportURL"><block s="reportJoinWords"><list><custom-block
s="$robot robot URL"></custom-block><l>/primitive/</l><block
var="primitive"/><l>/</l><block var="action"/></list></block></
block></block></script></block></script></block></script></block-
definition><block-definition s="$robot get %&apos;input&apos; of
behaviour %&apos;primitive&apos;" type="reporter"
category="sensing"><header></header><code></code><translations></
translations><inputs><input type="%s"
readonly="true">methodes<options>methodes&#xD;properties</options></
input><input type="%txt"></input></inputs><script><block
s="doDeclareVariables"><list><l>all primitives</l><l>keyword list</
```



```
l>/list></block><block s="doSetVar"><l>all primitives</l><custom-  
block s="$robot get %s behaviours"><l>all</l></custom-block></  
block><block s="doSetVar"><l>keyword list</l><block  
s="reportNewList"><list><l>properties</l><l>methodes</l></list></  
block></block><block s="doIfElse"><block s="reportNot"><block  
s="reportListContainsItem"><block var="all primitives"/><block  
var="primitive"/></block></block><script><custom-block s="$robot  
popup %txt"><block s="reportJoinWords"><list><l>Primitive </l><block  
var="primitive"/><l> is not in the aivaible primitives of your  
robot : </l><custom-block s="combine with %repRing items of  
%l"><block s="reifyReporter"><autolambda><custom-block s="join words  
%mult%txt"><list><l></l><l></l></list></custom-block></  
autolambda><list></list></block><block var="all primitives"/></  
custom-block></list></block></custom-block><block s="doReport"><l></  
l></block></script><script><block s="doIfElse"><block  
s="reportNot"><block s="reportListContainsItem"><block var="keyword  
list"/><block var="input"/></block></block><script><custom-block  
s="$robot popup %txt"><block s="reportJoinWords"><list><l>Input </  
l><block var="input"/><l> is not valid : </l><custom-block  
s="combine with %repRing items of %l"><block  
s="reifyReporter"><autolambda><custom-block s="join words  
%mult%txt"><list><l></l><l></l></list></custom-block></  
autolambda><list></list></block><block var="keyword list"/></custom-  
block></list></block></custom-block><block s="doReport"><l></l></  
block></script><script><block s="doReport"><block  
s="reportTextSplit"><block s="reportURL"><block  
s="reportJoinWords"><list><custom-block s="$robot robot URL"></  
custom-block><l>/primitive/</l><block var="primitive"/><l></  
l><block var="input"/></list></block></block><l></l></block></  
block></script></block></script></block></script></block-  
definition><block-definition s="set $robot host to  
'hostname'" type="command"  
category="variables"><header></header><code></code><translations></  
translations><inputs><input type="%s"></input></  
inputs><script><block s="doSetVar"><l>host</l><block var="hostname"/  
></block><block s="doDeclareVariables"><list><l>tmp_hostname</l></  
list></block><block s="doResetTimer"></block><block  
s="doUntil"><block s="reportGreaterThan"><block s="getTimer"></  
block><l>10</l></block><script><block s="doSetVar"><l>tmp_hostname</  
l><block s="reportURL"><block s="reportJoinWords"><list><custom-  
block s="$robot robot URL"></custom-block><l>/ip/</l></list></  
block></block></block><block s="doIf"><block  
s="reportGreaterThan"><block s="reportStringSize"><block  
var="tmp_hostname"/></block><l>1</l></block><script><block  
s="doSetVar"><l>host</l><block var="tmp_hostname"/></block><block  
s="bubble"><custom-block s="$robot test connection"></custom-  
block></block><block s="doReport"><l></l></block></script></  
block><block s="doWait"><l>0.1</l></block></script></block><custom-  
block s="$robot popup %txt"><l>Your robot host is unreacheble</l></  
custom-block></script></block-definition><block-definition s="ping  
http:// 'url'" type="predicate"  
category="operators"><header></header><code></code><translations></  
translations><inputs><input type="%s"></input></  
inputs><script><block s="doResetTimer"></block><block
```

```

s="doWaitUntil"><block s="report0r"><block
s="reportGreaterThan"><block s="getTimer"></block><l>5</l></
block><block s="reportGreaterThan"><block
s="reportStringSize"><block s="reportURL"><block var="url"/></
block></block><l>1</l></block></block></block><block
s="doIfElse"><block s="reportLessThan"><block s="getTimer"></
block><l>5</l></block><script><block s="doReport"><block
s="reportBoolean"><l><bool>true</bool></l></block></block></
script><script><block s="doReport"><block
s="reportBoolean"><l><bool>>false</bool></l></block></block></
script></block></script></block-definition><block-definition s="play
move %&apos;record_name&apos; | speed x %&apos;speed&apos;";
type="command" category="control"><comment x="0" y="0"
w="230.00000099999988" collapsed="false">Play the movement of a
previous recorded move. &#xD;&#xD;The &lt;record_name&gt; input is
the name of a .record filename which should be in the same directory
where you started the web server (with the command &quot;$robot-snap
$robot-ergo&quot; for example).&#xD;&#xD;You can change the play
speed with the &lt;speed&gt; input.</comment><header></
header><code></code><translations></translations><inputs><input
type="%txt">move_name</input><input type="%n">1</input></
inputs><script><block s="doDeclareVariables"><list><l>time</l></
list></block><block s="doDeclareVariables"><list><l>wait</l></
list></block><block s="doSetVar"><l>wait</l><block
s="reportBoolean"><l><bool>true</bool></l></block></block><block
s="doIfElse"><block s="reportListContainsItem"><custom-block
s="$robot all recorded moves"></custom-block><block
var="record_name"/></block><script><block s="doSetVar"><l>time</
l><block s="reportURL"><block s="reportJoinWords"><list><custom-
block s="$robot robot URL"></custom-block><l>/primitive/MovePlayer/
</l><block var="record_name"/><l>/start/</l><block var="speed"/></
list></block></block></block><block s="doIf"><block var="wait"/
><script><block s="doWait"><block var="time"/></block></
script><comment w="90" collapsed="false">ugly wait to stop
feature.&#xD;TODO : change with a looking at running primitive</
comment></block></script><script><custom-block s="$robot popup
%txt"><block s="reportJoinWords"><list><l>Error : &lt;</l><block
var="record_name"/><l>&gt; is not is the avabile records (look at
the block $robot all records to see all recorded moves) </l><custom-
block s="$robot all recorded moves"></custom-block></list></block></
custom-block></script></block><block s="doReport"><block
var="record_name"/></block></script></block-definition><block-
definition s="$robot set color leds of motor(s)
&apos;motor_name&apos; in %&apos;color&apos;" type="command"
category="control"><header></header><code></code><translations></
translations><inputs><input type="%s">motor_name</input><input
type="%txt">off<options>off&#xD;red&#xD;green&#xD;yellow&#xD;blue&#x
D;pink&#xD;cyan&#xD;white</options></input></inputs><script><custom-
block s="$robot set %txt of motor(s) %s to %s"><l>led</l><block
var="motor_name"/><block var="color"/></custom-block></script></
block-definition><block-definition s="$robot card %&apos;card&apos;
is detected ?" type="predicate" category="sensing"><header></
header><code></code><translations></translations><inputs><input
type="%txt">caribou<options>caribou&#xD;tetris&#xD;lapin</options></

```

```
input></inputs><script><block
s="doDeclareVariables"><list><l>output</l></list></block><block
s="doSetVar"><l>output</l><block s="reportURL"><block
s="reportJoinWords"><list><custom-block s="$robot robot URL"></
custom-block><l>/detect/</l><block var="card"/></list></block></
block></block><block s="doIfElse"><block s="reportEquals"><block
var="output"/><l>True</l></block><script><block s="doReport"><block
s="reportBoolean"><l><bool>>true</bool></l></block></block></
script><script><block s="doReport"><block
s="reportBoolean"><l><bool>>false</bool></l></block></block></
script></block></script></block-definition></
blocks><variables><variable name="host"><l>192.168.0.33</l></
variable><variable name="port"><l>6969</l></variable><variable
name="ang"><l>-49</l></variable></variables></project>
```