Code for the English version of the approach avoidance task on Inquisit 6 (Millisecond Software, Version 6.5.2, <https://www.millisecond.com>)

<usermanual>

LIFESPAN STUDY (ENGLISH VERSION)

</usermanual>

<parameters>

/ responsekey\_up = 22

/ responsekey\_down = 49

/ upkeylabel = "U"

/ downkeylabel = "N"

</parameters>

<instruct>

/ fontstyle = ("Arial", 3%, false, false, false, false, 5, 1)

/ txcolor = black

/ lastlabel = ("Press the space bar to continue")

/ nextlabel = ("Press the space bar to continue")

</instruct>

<defaults >

/canvasaspectratio = (4,3)

/ minimumversion = "6.5.0.0"

/ fontstyle = ("Arial", 3%, false, false, false, false, 5, 1)

/txbgcolor = white

/ txcolor = (0, 0, 0)

/ inputdevice = keyboard

</defaults>

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Output of raw data

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

<data>

/ separatefiles = true

/ columns = (textbox.participant.response, build, computer.platform, date, time, subject, group, blockcode, blocknum, trialcode, latency, correct, trialnum, response, script.elapsedtime,

parameters.responsekey\_up, parameters.responsekey\_down, values.ap\_pictureindex, values.seden\_pictureindex, values.circle\_pictureindex, values.square\_pictureindex,textbox.age.response,textbox.height.response,textbox.weight.response,radiobuttons.sex.response,radiobuttons.gender.response,textbox.country.response,checkboxes.chronic.response,

textbox.vigorous\_d.response,textbox.vigorous\_m.response,textbox.moderate\_d.response,textbox.moderate\_m.response,textbox.sedentary\_d.response,textbox.sedentary\_m.response,radiobuttons.intention.response,radiobuttons.attitude1.response,radiobuttons.attitude2.response,

expressions.propcorr\_ApApproach, expressions.meanRT\_ApApproach,

expressions.propcorr\_ApAvoid, expressions.meanRT\_ApAvoid,

expressions.propcorr\_SedenApproach, expressions.meanRT\_SedenApproach,

expressions.propcorr\_SedenAvoid, expressions.meanRT\_SedenAvoid,

expressions.propcorr\_circleApproach, expressions.meanRT\_circleApproach,

expressions.propcorr\_circleAvoid, expressions.meanRT\_circleAvoid,

expressions.propcorr\_squareApproach, expressions.meanRT\_squareApproach,

expressions.propcorr\_squareAvoid, expressions.meanRT\_squareAvoid)

</data>

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Approach-Avoidance Task

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

<values>

/ completed = 0

/ ap\_pictureindex = 0

/ seden\_pictureindex = 0

/ circle\_pictureindex = 0

/ square\_pictureindex = 0

/ nextpicture = 0

/ selectap = 0

/ selectseden = 0

/ stimulus = ""

/ test = 0

</values>

\*\*\*\*\*

<expressions >

/ propcorr\_total = list.accuracy\_alltrials.mean

/ meanRT\_total = list.latencies\_allcorrecttrials.mean

/ propcorr\_ApApproach = list.accuracy\_ApproachAp.mean

/ meanRT\_ApApproach = list.latencies\_corrApproachAp.mean

/ propcorr\_ApAvoid = list.accuracy\_AvoidAp.mean

/ meanRT\_ApAvoid = list.latencies\_corrAvoidAp.mean

/ propcorr\_SedenApproach = list.accuracy\_ApproachSeden.mean

/ meanRT\_SedenApproach = list.latencies\_corrApproachSeden.mean

/ propcorr\_SedenAvoid = list.accuracy\_AvoidSeden.mean

/ meanRT\_SedenAvoid = list.latencies\_corrAvoidSeden.mean

/ propcorr\_circleApproach = list.accuracy\_ApproachCircle.mean

/ meanRT\_circleApproach = list.latencies\_corrApproachCircle.mean

/ propcorr\_circleAvoid = list.accuracy\_AvoidCircle.mean

/ meanRT\_circleAvoid = list.latencies\_corrAvoidCircle.mean

/ propcorr\_squareApproach = list.accuracy\_ApproachSquare.mean

/ meanRT\_squareApproach = list.latencies\_corrApproachSquare.mean

/ propcorr\_squareAvoid = list.accuracy\_AvoidSquare.mean

/ meanRT\_squareAvoid = list.latencies\_corrAvoidSquare.mean

</expressions>

<expressions>

/fixduration = rand(500,750)

</expressions>

<picture ap>

/ items = appics

/ position = (50%,50%)

/ erase = false

</picture>

<picture seden>

/ items = sedenpics

/ position = (50%,50%)

/ erase = false

</picture>

<picture circle>

/ items = circlepics

/ position = (50%,50%)

/ size = (20%, 20%)

/ erase = false

</picture>

<picture square>

/ items = squarepics

/ position = (50%,50%)

/ size = (20%, 20%)

/ erase = false

</picture>

\*\*\*\*\*\*\*\*\*

<page intro>

^Thank you for participating in this online study that is divided into 3 parts:

^^1- Informed consent and demographic questions.

^^2- The "Manikin task" in 4 different conditions.

^^3- Questionnaires.

^^You need to answer all the questions on the page to be able to move to the next page by pressing the "Next" button located at the bottom of the page.

^^Please press the space bar on your keyboard when you are ready to start.

</page>

<page end>

^^Thank you for your participation :)

</page>

\*\*\*Manikin instructions

<page approach\_circle>

^^Please locate the"<%parameters.upkeylabel%>" and "<%parameters.downkeylabel%>" keys on your keyboard. These 2 keys will be useful here.

^^In the upcoming task, a manikin that represents YOU will appear either at the top of the screen, or at the bottom.

^^Then, an image will appear in the center of the screen.

^^Your task is to move the manikin as quickly as possible according to what the image depicts.

^^ - If the image depicts CIRCLES and OVALS, you should move the manikin TOWARDS that circle (APPROACH CIRCLE).

^^ - If the image depicts SQUARES and RECTANGLES, you should move the manikin AWAY from that square (AVOID SQUARE).

^^With your index fingers, press the keyboard key "<%parameters.upkeylabel%>" to move the manikin up and the key "<%parameters.downkeylabel%>" to move it down.

^^Please press the SPACE BAR to start the session.

</page>

<page approach\_ap>

^^Please locate the"<%parameters.upkeylabel%>" and "<%parameters.downkeylabel%>" keys on your keyboard. These 2 keys will be useful here.

^^In the upcoming task, a manikin that represents YOU will appear either at the top of the screen, or at the bottom.

^^Then, an image will appear in the center of the screen.

^^Your task is to move the manikin as quickly as possible according to what the image depicts.

^^ - If the image depicts a PHYSICAL ACTIVITY, you should move the manikin TOWARDS that image (APPROACH PHYSICAL ACTIVITY).

^^ - If the image depicts a SEDENTARY BEHAVIOR, you should move the manikin AWAY from that image (AVOID SEDENTARY BEHAVIOR).

^^With your index fingers, press the keyboard key "<%parameters.upkeylabel%>" to move the manikin up and the key "<%parameters.downkeylabel%>" to move it down.

^^Please press the SPACE BAR to start the session.

</page>

<page avoid\_circle>

^^Please locate the"<%parameters.upkeylabel%>" and "<%parameters.downkeylabel%>" keys on your keyboard. These 2 keys will be useful here.

^^In the upcoming task, a manikin that represents YOU will appear either at the top of the screen, or at the bottom.

^^Then, an image will appear in the center of the screen.

^^Your task is to move the manikin as quickly as possible according to what the image depicts.

^^ - If the image depicts CIRCLES and OVALS, you should move the manikin AWAY from that circle (AVOID CIRCLE).

^^ - If the image depicts SQUARES and RECTANGLES, you should move the manikin TOWARDS that square (APPROACH SQUARE).

^^With your index fingers, press the keyboard key "<%parameters.upkeylabel%>" to move the manikin up and the key "<%parameters.downkeylabel%>" to move it down.

^^Please press the SPACE BAR to start the session.

</page>

<page avoid\_ap>

^^Please locate the"<%parameters.upkeylabel%>" and "<%parameters.downkeylabel%>" keys on your keyboard. These 2 keys will be useful here.

^^In the upcoming task, a manikin that represents YOU will appear either at the top of the screen, or at the bottom.

^^Then, an image will appear in the center of the screen.

^^Your task is to move the manikin as quickly as possible according to what the image depicts.

^^ - If the image depicts a PHYSICAL ACTIVITY, you should move the manikin AWAY from that image (AVOID PHYSICAL ACTIVITY).

^^ - If the image depicts a SEDENTARY BEHAVIOR, you should move the manikin TOWARDS that image (APPROACH SEDENTARY BEHAVIOR).

^^With your index fingers, press the keyboard key "<%parameters.upkeylabel%>" to move the manikin up and the key "<%parameters.downkeylabel%>" to move it down.

^^Please press the SPACE BAR to start the session.

</page>

........

<text fixation>

/ items = ("+")

/ fontstyle = ("Arial", 5.00%, true)

</text>

<text reminder>

/ items = ("Congratulations on completing this part of the study.

Please read the following message carefully as the instructions change.

~nPlease press the space bar to continue")

/ fontstyle = ("Arial", 3.00%, true)

</text>

<picture manikintop>

/ items = ("manikin\_2.jpg")

/ position = (50%, 20%)

</picture>

<picture manikin\_top\_moveup>

/ items = ("manikin\_2.jpg")

/ position = (50%, 10%)

</picture>

<picture manikin\_top\_movedown>

/ items = ("manikin\_2.jpg")

/ position = (50%, 30%)

</picture>

<picture manikinbottom>

/ items = ("manikin\_2.jpg")

/ position = (50%, 80%)

</picture>

<picture manikin\_bottom\_moveup>

/ items = ("manikin\_2.jpg")

/ position = (50%, 70%)

</picture>

<picture manikin\_bottom\_movedown>

/ items = ("manikin\_2.jpg")

/ position = (50%, 90%)

</picture>

<text error>

/ fontstyle = ("arial", 22pt, true)

/ items = ("ERROR")

/ position = (50%,50%)

/ txcolor = (red)

/ erase = false

</text>

<text trop\_lent>

/ fontstyle = ("arial", 22pt, true)

/ items = ("TOO SLOW")

/ position = (50%,50%)

/ txcolor = (red)

/ erase = false

</text>

<picture instructionimages>

/ items = ("instructionimages.png")

/ position = (50%,50%)

/ size = (80%, 80%)

</picture>

Note: the following 4 lists help determine which pictures are presented during the experimental trials

<list ap\_pictureindex>

/ items = (1, 2, 3, 4, 5, 6)

/ selectionrate = trial

/ selectionmode = random

</list>

<list seden\_pictureindex>

/ items = (7, 8, 9, 10, 11, 12)

/ selectionrate = trial

/ selectionmode = random

</list>

<list circle\_pictureindex>

/ items = (13,14,15,16,17,18)

/ selectionrate = trial

/ selectionmode = random

</list>

<list square\_pictureindex>

/ items = (19,20,21,22,23,24)

/ selectionrate = trial

/ selectionmode = random

</list>

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Data Lists: used for descriptive statistics

store latencies/accuracy data

fill up during runtime

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

<list accuracy\_alltrials>

</list>

<list latencies\_allcorrecttrials>

</list>

<list accuracy\_ApproachAp>

</list>

<list latencies\_corrApproachAp>

</list>

<list accuracy\_AvoidAp>

</list>

<list latencies\_corrAvoidAp>

</list>

<list accuracy\_ApproachSeden>

</list>

<list latencies\_corrApproachSeden>

</list>

<list accuracy\_AvoidSeden>

</list>

<list latencies\_corrAvoidSeden>

</list>

<list accuracy\_ApproachCircle>

</list>

<list latencies\_corrApproachCircle>

</list>

<list accuracy\_AvoidCircle>

</list>

<list latencies\_corrAvoidCircle>

</list>

<list accuracy\_ApproachSquare>

</list>

<list latencies\_corrApproachSquare>

</list>

<list accuracy\_AvoidSquare>

</list>

<list latencies\_corrAvoidSquare>

</list>

............

<item appics>

/ 1 = "AP-COUR.jpg"

/ 2 = "AP-ESCAL.jpg"

/ 3 = "AP-FOOT.jpg"

/ 4 = "AP-NAT.jpg"

/ 5 = "AP-RANDO.jpg"

/ 6 = "AP-VEL.jpg"

</item>

<item sedenpics>

/ 7 = "SED-CANAP.jpg"

/ 8 = "SED-HAMAC.jpg"

/ 9 = "SED-JVID.jpg"

/ 10 = "SED-LECT.jpg"

/ 11 = "SED-ORDI.jpg"

/ 12 = "SED-TV.jpg"

</item>

<item circlepics>

/ 13 = "AP-NATr.jpg"

/ 14 = "AP-RANDOr.jpg"

/ 15 = "AP-VELr.jpg"

/ 16 = "SED-CANAPr.jpg"

/ 17 = "SED-HAMACr.jpg"

/ 18 = "SED-LECTr.jpg"

</item>

<item squarepics>

/ 19 = "AP-NATc.jpg"

/ 20 = "AP-RANDOc.jpg"

/ 21 = "AP-VELc.jpg"

/ 22 = "SED-CANAPc.jpg"

/ 23 = "SED-HAMACc.jpg"

/ 24 = "SED-LECTc.jpg"

</item>

........

........

<trial error>

/ trialcode = "error"

/ stimulustimes = [0= error]

/ trialduration = (800)

/ responsemode = noresponse

</trial>

<trial trop\_lent>

/ trialcode = "too\_slow"

/ stimulustimes = [0= trop\_lent]

/ trialduration = (800)

/ responsemode = noresponse

</trial>

<trial instructionimages>

/trialcode = "instructionimages"

/ stimulustimes = [0=instructionimages]

/ trialduration = 7000

</trial>

\*\*\*\*\*\*\*physical activity picture man top approach: 1

<trial ApApproach\_ManikinTop>

/ ontrialbegin = [values.ap\_pictureindex = list.ap\_pictureindex.nextvalue]

/ stimulustimes = [0 = clearscreen, manikintop; 1000=ap]

/ validresponse = (parameters.responsekey\_up, parameters.responsekey\_down)

/ correctresponse = (parameters.responsekey\_down)

/ ontrialend = [

if (values.test == 1) {

list.accuracy\_alltrials.insertitem(trial.ApApproach\_ManikinTop.correct, 1);

list.accuracy\_ApproachAp.insertitem(trial.ApApproach\_ManikinTop.correct, 1);

if (trial.ApApproach\_ManikinTop.correct) {

list.latencies\_allcorrecttrials.insertitem(trial.ApApproach\_ManikinTop.latency, 1);

list.latencies\_corrApproachAp.insertitem(trial.ApApproach\_ManikinTop.latency, 1);

}

}

]

/ correctmessage = (manikin\_top\_movedown, 1000)

/ branch = [if (trial.ApApproach\_ManikinTop.response == 0) trial.trop\_lent]

/ branch = [if (trial.ApApproach\_ManikinTop.error) trial.error]

/ timeout = 8000

</trial>

\*\*\*\*\*\*\*physical activity picture man bottom approach: 2

<trial ApApproach\_ManikinBottom>

/ ontrialbegin = [values.ap\_pictureindex = list.ap\_pictureindex.nextvalue]

/ stimulustimes = [0 = clearscreen, manikinbottom; 1000=ap]

/ validresponse = (parameters.responsekey\_up, parameters.responsekey\_down)

/ correctresponse = (parameters.responsekey\_up)

/ ontrialend = [

if (values.test == 1) {

list.accuracy\_alltrials.insertitem(trial.ApApproach\_ManikinBottom.correct, 1);

list.accuracy\_ApproachAp.insertitem(trial.ApApproach\_ManikinBottom.correct, 1);

if (trial.ApApproach\_ManikinBottom.correct) {

list.latencies\_allcorrecttrials.insertitem(trial.ApApproach\_ManikinBottom.latency, 1);

list.latencies\_corrApproachAp.insertitem(trial.ApApproach\_ManikinBottom.latency, 1);

}

}

]

/ branch = [if (trial.ApApproach\_ManikinBottom.response == 0) trial.trop\_lent]

/ branch = [if (trial.ApApproach\_ManikinBottom.error) trial.error]

/ correctmessage = (manikin\_bottom\_moveup, 1000)

/ timeout = 8000

</trial>

\*\*\*\*\*\*\*physical activity picture man top avoid: 3

<trial ApAvoid\_ManikinTop>

/ ontrialbegin = [values.ap\_pictureindex = list.ap\_pictureindex.nextvalue]

/ stimulustimes = [0 = clearscreen, manikintop; 1000=ap]

/ validresponse = (parameters.responsekey\_up, parameters.responsekey\_down)

/ correctresponse = (parameters.responsekey\_up)

/ ontrialend = [

if (values.test == 1) {

list.accuracy\_alltrials.insertitem(trial.ApAvoid\_ManikinTop.correct, 1);

list.accuracy\_AvoidAp.insertitem(trial.ApAvoid\_ManikinTop.correct, 1);

if (trial.ApAvoid\_ManikinTop.correct) {

list.latencies\_allcorrecttrials.insertitem(trial.ApAvoid\_ManikinTop.latency, 1);

list.latencies\_corrAvoidAp.insertitem(trial.ApAvoid\_ManikinTop.latency, 1);

}

}

]

/ branch = [if (trial.ApAvoid\_ManikinTop.response == 0) trial.trop\_lent]

/ branch = [if (trial.ApAvoid\_ManikinTop.error) trial.error]

/ correctmessage = (manikin\_top\_moveup, 1000)

/ timeout = 8000

</trial>

\*\*\*\*\*\*\*physical activity picture man bottom avoid: 4

<trial ApAvoid\_ManikinBottom>

/ ontrialbegin = [values.ap\_pictureindex = list.ap\_pictureindex.nextvalue]

/ stimulustimes = [0 = clearscreen, manikinbottom; 1000=ap]

/ validresponse = (parameters.responsekey\_up, parameters.responsekey\_down)

/ correctresponse = (parameters.responsekey\_down)

/ ontrialend = [

if (values.test == 1) {

list.accuracy\_alltrials.insertitem(trial.ApAvoid\_ManikinBottom.correct, 1);

list.accuracy\_AvoidAp.insertitem(trial.ApAvoid\_ManikinBottom.correct, 1);

if (trial.ApAvoid\_ManikinBottom.correct) {

list.latencies\_allcorrecttrials.insertitem(trial.ApAvoid\_ManikinBottom.latency, 1);

list.latencies\_corrAvoidAp.insertitem(trial.ApAvoid\_ManikinBottom.latency, 1);

}

}

]

/ branch = [if (trial.ApAvoid\_ManikinBottom.response == 0) trial.trop\_lent]

/ branch = [if (trial.ApAvoid\_ManikinBottom.error) trial.error]

/ correctmessage = (manikin\_bottom\_movedown, 1000)

/ timeout = 8000

</trial>

\*\*\*\*\*\*\*sedentary picture man top approach: 5

<trial SedenApproach\_ManikinTop>

/ ontrialbegin = [values.seden\_pictureindex = list.seden\_pictureindex.nextvalue]

/ stimulustimes = [0 = clearscreen, manikintop; 1000=seden]

/ validresponse = (parameters.responsekey\_up, parameters.responsekey\_down)

/ correctresponse = (parameters.responsekey\_down)

/ ontrialend = [

if (values.test == 1) {

list.accuracy\_alltrials.insertitem(trial.SedenApproach\_ManikinTop.correct, 1);

list.accuracy\_ApproachSeden.insertitem(trial.SedenApproach\_ManikinTop.correct, 1);

if (trial.SedenApproach\_ManikinTop.correct) {

list.latencies\_allcorrecttrials.insertitem(trial.SedenApproach\_ManikinTop.latency, 1);

list.latencies\_corrApproachSeden.insertitem(trial.SedenApproach\_ManikinTop.latency, 1);

}

}

]

/ branch = [if (trial.SedenApproach\_ManikinTop.response == 0) trial.trop\_lent]

/ branch = [if (trial.SedenApproach\_ManikinTop.error) trial.error]

/ correctmessage = (manikin\_top\_movedown, 1000)

/ timeout = 8000

</trial>

\*\*\*\*\*\*\*sedentary picture man bottom approach: 6

<trial SedenApproach\_ManikinBottom>

/ ontrialbegin = [values.seden\_pictureindex = list.seden\_pictureindex.nextvalue]

/ stimulustimes = [0=clearscreen, manikinbottom; 1000=seden]

/ validresponse = (parameters.responsekey\_up, parameters.responsekey\_down)

/ correctresponse = (parameters.responsekey\_up)

/ ontrialend = [

if (values.test == 1) {

list.accuracy\_alltrials.insertitem(trial.SedenApproach\_ManikinBottom.correct, 1);

list.accuracy\_ApproachSeden.insertitem(trial.SedenApproach\_ManikinBottom.correct, 1);

if (trial.SedenApproach\_ManikinBottom.correct) {

list.latencies\_allcorrecttrials.insertitem(trial.SedenApproach\_ManikinBottom.latency, 1);

list.latencies\_corrApproachSeden.insertitem(trial.SedenApproach\_ManikinBottom.latency, 1);

}

}

]

/ branch = [if (trial.SedenApproach\_ManikinBottom.response == 0) trial.trop\_lent]

/ branch = [if (trial.SedenApproach\_ManikinBottom.error) trial.error]

/ correctmessage = (manikin\_bottom\_moveup, 1000)

/ timeout = 8000

</trial>

\*\*\*\*\*\*\*sedentaryt picture man top avoid: 7

<trial SedenAvoid\_ManikinTop>

/ ontrialbegin = [values.seden\_pictureindex = list.seden\_pictureindex.nextvalue]

/ stimulustimes = [0 = clearscreen, manikintop; 1000=seden]

/ validresponse = (parameters.responsekey\_up, parameters.responsekey\_down)

/ correctresponse = (parameters.responsekey\_up)

/ ontrialend = [

if (values.test == 1) {

list.accuracy\_alltrials.insertitem(trial.SedenAvoid\_ManikinTop.correct, 1);

list.accuracy\_AvoidSeden.insertitem(trial.SedenAvoid\_ManikinTop.correct, 1);

if (trial.SedenAvoid\_ManikinTop.correct) {

list.latencies\_allcorrecttrials.insertitem(trial.SedenAvoid\_ManikinTop.latency, 1);

list.latencies\_corrAvoidSeden.insertitem(trial.SedenAvoid\_ManikinTop.latency, 1);

}

}

]

/ branch = [if (trial.SedenAvoid\_ManikinTop.response == 0) trial.trop\_lent]

/ branch = [if (trial.SedenAvoid\_ManikinTop.error) trial.error]

/ correctmessage = (manikin\_top\_moveup, 1000)

/ timeout = 8000

</trial>

\*\*\*\*\*\*\*sedentary picture man bottom avoid: 8

<trial SedenAvoid\_ManikinBottom>

/ ontrialbegin = [values.seden\_pictureindex = list.seden\_pictureindex.nextvalue]

/ stimulustimes = [0=clearscreen, manikinbottom; 1000=seden]

/ validresponse = (parameters.responsekey\_up, parameters.responsekey\_down)

/ correctresponse = (parameters.responsekey\_down)

/ ontrialend = [

if (values.test == 1) {

list.accuracy\_alltrials.insertitem(trial.SedenAvoid\_ManikinBottom.correct, 1);

list.accuracy\_AvoidSeden.insertitem(trial.SedenAvoid\_ManikinBottom.correct, 1);

if (trial.SedenAvoid\_ManikinBottom.correct) {

list.latencies\_allcorrecttrials.insertitem(trial.SedenAvoid\_ManikinBottom.latency, 1);

list.latencies\_corrAvoidSeden.insertitem(trial.SedenAvoid\_ManikinBottom.latency, 1);

}

}

]

/ branch = [if (trial.SedenAvoid\_ManikinBottom.response == 0) trial.trop\_lent]

/ branch = [if (trial.SedenAvoid\_ManikinBottom.error) trial.error]

/ correctmessage = (manikin\_bottom\_movedown, 1000)

/ timeout = 8000

</trial>

\*\*\*\*\*\*\*circle picture man top approach: 1

<trial circleApproach\_ManikinTop>

/ ontrialbegin = [values.circle\_pictureindex = list.circle\_pictureindex.nextvalue]

/ stimulustimes = [0 = clearscreen, manikintop; 1000=circle]

/ validresponse = (parameters.responsekey\_up, parameters.responsekey\_down)

/ correctresponse = (parameters.responsekey\_down)

/ ontrialend = [

if (values.test == 1) {

list.accuracy\_alltrials.insertitem(trial.circleApproach\_ManikinTop.correct, 1);

list.accuracy\_ApproachCircle.insertitem(trial.circleApproach\_ManikinTop.correct, 1);

if (trial.circleApproach\_ManikinTop.correct) {

list.latencies\_allcorrecttrials.insertitem(trial.circleApproach\_ManikinTop.latency, 1);

list.latencies\_corrApproachCircle.insertitem(trial.circleApproach\_ManikinTop.latency, 1);

}

}

]

/ correctmessage = (manikin\_top\_movedown, 1000)

/ branch = [if (trial.circleApproach\_ManikinTop.response == 0) trial.trop\_lent]

/ branch = [if (trial.circleApproach\_ManikinTop.error) trial.error]

/ timeout = 8000

</trial>

\*\*\*\*\*\*\*circle picture man bottom approach: 2

<trial circleApproach\_ManikinBottom>

/ ontrialbegin = [values.circle\_pictureindex = list.circle\_pictureindex.nextvalue]

/ stimulustimes = [0 = clearscreen, manikinbottom; 1000=circle]

/ validresponse = (parameters.responsekey\_up, parameters.responsekey\_down)

/ correctresponse = (parameters.responsekey\_up)

/ ontrialend = [

if (values.test == 1) {

list.accuracy\_alltrials.insertitem(trial.circleApproach\_ManikinBottom.correct, 1);

list.accuracy\_ApproachCircle.insertitem(trial.circleApproach\_ManikinBottom.correct, 1);

if (trial.circleApproach\_ManikinBottom.correct) {

list.latencies\_allcorrecttrials.insertitem(trial.circleApproach\_ManikinBottom.latency, 1);

list.latencies\_corrApproachCircle.insertitem(trial.circleApproach\_ManikinBottom.latency, 1);

}

}

]

/ branch = [if (trial.circleApproach\_ManikinBottom.response == 0) trial.trop\_lent]

/ branch = [if (trial.circleApproach\_ManikinBottom.error) trial.error]

/ correctmessage = (manikin\_bottom\_moveup, 1000)

/ timeout = 8000

</trial>

\*\*\*\*\*\*\*circle picture man top avoid: 3

<trial circleAvoid\_ManikinTop>

/ ontrialbegin = [values.circle\_pictureindex = list.circle\_pictureindex.nextvalue]

/ stimulustimes = [0 = clearscreen, manikintop; 1000=circle]

/ validresponse = (parameters.responsekey\_up, parameters.responsekey\_down)

/ correctresponse = (parameters.responsekey\_up)

/ ontrialend = [

if (values.test == 1) {

list.accuracy\_alltrials.insertitem(trial.circleAvoid\_ManikinTop.correct, 1);

list.accuracy\_AvoidCircle.insertitem(trial.circleAvoid\_ManikinTop.correct, 1);

if (trial.circleAvoid\_ManikinTop.correct) {

list.latencies\_allcorrecttrials.insertitem(trial.circleAvoid\_ManikinTop.latency, 1);

list.latencies\_corrAvoidCircle.insertitem(trial.circleAvoid\_ManikinTop.latency, 1);

}

}

]

/ branch = [if (trial.circleAvoid\_ManikinTop.response == 0) trial.trop\_lent]

/ branch = [if (trial.circleAvoid\_ManikinTop.error) trial.error]

/ correctmessage = (manikin\_top\_moveup, 1000)

/ timeout = 8000

</trial>

\*\*\*\*\*\*\*circle picture man bottom avoid: 4

<trial circleAvoid\_ManikinBottom>

/ ontrialbegin = [values.circle\_pictureindex = list.circle\_pictureindex.nextvalue]

/ stimulustimes = [0 = clearscreen, manikinbottom; 1000=circle]

/ validresponse = (parameters.responsekey\_up, parameters.responsekey\_down)

/ correctresponse = (parameters.responsekey\_down)

/ ontrialend = [

if (values.test == 1) {

list.accuracy\_alltrials.insertitem(trial.circleAvoid\_ManikinBottom.correct, 1);

list.accuracy\_AvoidCircle.insertitem(trial.circleAvoid\_ManikinBottom.correct, 1);

if (trial.circleAvoid\_ManikinBottom.correct) {

list.latencies\_allcorrecttrials.insertitem(trial.circleAvoid\_ManikinBottom.latency, 1);

list.latencies\_corrAvoidCircle.insertitem(trial.circleAvoid\_ManikinBottom.latency, 1);

}

}

]

/ branch = [if (trial.circleAvoid\_ManikinBottom.response == 0) trial.trop\_lent]

/ branch = [if (trial.circleAvoid\_ManikinBottom.error) trial.error]

/ correctmessage = (manikin\_bottom\_movedown, 1000)

/ timeout = 8000

</trial>

\*\*\*\*\*\*\*square picture man top approach: 5

<trial squareApproach\_ManikinTop>

/ ontrialbegin = [values.square\_pictureindex = list.square\_pictureindex.nextvalue]

/ stimulustimes = [0 = clearscreen, manikintop; 1000=square]

/ validresponse = (parameters.responsekey\_up, parameters.responsekey\_down)

/ correctresponse = (parameters.responsekey\_down)

/ ontrialend = [

if (values.test == 1) {

list.accuracy\_alltrials.insertitem(trial.squareApproach\_ManikinTop.correct, 1);

list.accuracy\_ApproachSquare.insertitem(trial.squareApproach\_ManikinTop.correct, 1);

if (trial.squareApproach\_ManikinTop.correct) {

list.latencies\_allcorrecttrials.insertitem(trial.squareApproach\_ManikinTop.latency, 1);

list.latencies\_corrApproachSquare.insertitem(trial.squareApproach\_ManikinTop.latency, 1);

}

}

]

/ branch = [if (trial.squareApproach\_ManikinTop.response == 0) trial.trop\_lent]

/ branch = [if (trial.squareApproach\_ManikinTop.error) trial.error]

/ correctmessage = (manikin\_top\_movedown, 1000)

/ timeout = 8000

</trial>

\*\*\*\*\*\*\*square picture man bottom approach: 6

<trial squareApproach\_ManikinBottom>

/ ontrialbegin = [values.square\_pictureindex = list.square\_pictureindex.nextvalue]

/ stimulustimes = [0=clearscreen, manikinbottom; 1000=square]

/ validresponse = (parameters.responsekey\_up, parameters.responsekey\_down)

/ correctresponse = (parameters.responsekey\_up)

/ ontrialend = [

if (values.test == 1) {

list.accuracy\_alltrials.insertitem(trial.squareApproach\_ManikinBottom.correct, 1);

list.accuracy\_ApproachSquare.insertitem(trial.squareApproach\_ManikinBottom.correct, 1);

if (trial.squareApproach\_ManikinBottom.correct) {

list.latencies\_allcorrecttrials.insertitem(trial.squareApproach\_ManikinBottom.latency, 1);

list.latencies\_corrApproachSquare.insertitem(trial.squareApproach\_ManikinBottom.latency, 1);

}

}

]

/ branch = [if (trial.squareApproach\_ManikinBottom.response == 0) trial.trop\_lent]

/ branch = [if (trial.squareApproach\_ManikinBottom.error) trial.error]

/ correctmessage = (manikin\_bottom\_moveup, 1000)

/ timeout = 8000

</trial>

\*\*\*\*\*\*\*square picture man top avoid: 7

<trial squareAvoid\_ManikinTop>

/ ontrialbegin = [values.square\_pictureindex = list.square\_pictureindex.nextvalue]

/ stimulustimes = [0 = clearscreen, manikintop; 1000=square]

/ validresponse = (parameters.responsekey\_up, parameters.responsekey\_down)

/ correctresponse = (parameters.responsekey\_up)

/ ontrialend = [

if (values.test == 1) {

list.accuracy\_alltrials.insertitem(trial.squareAvoid\_ManikinTop.correct, 1);

list.accuracy\_AvoidSquare.insertitem(trial.squareAvoid\_ManikinTop.correct, 1);

if (trial.squareAvoid\_ManikinTop.correct) {

list.latencies\_allcorrecttrials.insertitem(trial.squareAvoid\_ManikinTop.latency, 1);

list.latencies\_corrAvoidSquare.insertitem(trial.squareAvoid\_ManikinTop.latency, 1);

}

}

]

/ branch = [if (trial.squareAvoid\_ManikinTop.response == 0) trial.trop\_lent]

/ branch = [if (trial.squareAvoid\_ManikinTop.error) trial.error]

/ correctmessage = (manikin\_top\_moveup, 1000)

/ timeout = 8000

</trial>

\*\*\*\*\*\*\*square picture man bottom avoid: 8

<trial squareAvoid\_ManikinBottom>

/ ontrialbegin = [values.square\_pictureindex = list.square\_pictureindex.nextvalue]

/ stimulustimes = [0=clearscreen, manikinbottom; 1000=square]

/ validresponse = (parameters.responsekey\_up, parameters.responsekey\_down)

/ correctresponse = (parameters.responsekey\_down)

/ ontrialend = [

if (values.test == 1) {

list.accuracy\_alltrials.insertitem(trial.squareAvoid\_ManikinBottom.correct, 1);

list.accuracy\_AvoidSquare.insertitem(trial.squareAvoid\_ManikinBottom.correct, 1);

if (trial.squareAvoid\_ManikinBottom.correct) {

list.latencies\_allcorrecttrials.insertitem(trial.squareAvoid\_ManikinBottom.latency, 1);

list.latencies\_corrAvoidSquare.insertitem(trial.squareAvoid\_ManikinBottom.latency, 1);

}

}

]

/ branch = [if (trial.squareAvoid\_ManikinBottom.response == 0) trial.trop\_lent]

/ branch = [if (trial.squareAvoid\_ManikinBottom.error) trial.error]

/ correctmessage = (manikin\_bottom\_movedown, 1000)

/ timeout = 8000

</trial>

<trial fixation>

/stimulustimes = [0=clearscreen, fixation]

/trialduration = (expressions.fixduration)

/correctresponse = (noresponse)

</trial>

<trial reminder>

/stimulustimes = [0=clearscreen, reminder]

/ correctresponse = (57)

</trial>

\*\*\* 48 trials per condition. The Fixation command makes the fixation cross appear on the screen.

<block approach\_circle>

/ onblockbegin = [values.test = 1;]

/ screencolor = white

/ preinstructions = (approach\_circle)

/ trials = [1,3,5,7,9,11,13,15,17,19,21,23,25,27,29,31,33,35,37,39,41,43,45,47,49,51,53,55,57,59,61,63,65,67,69,71,73,75,77,79,81,83,85,87,89,91,93,95 = fixation;

2,4,6,8,10,12,14,16,18,20,22,24,26,28,30,32,34,36,38,40,42,44,46,48,50,52,54,56,58,60,62,64,66,68,70,72,74,76,78,80,82,84,86,88,90,92,94,96 = noreplace (circleApproach\_ManikinTop,circleApproach\_ManikinBottom, squareAvoid\_ManikinTop, squareAvoid\_ManikinBottom)]

</block>

<block avoid\_circle>

/ onblockbegin = [values.test = 1;]

/ screencolor = white

/ preinstructions = (avoid\_circle)

/ trials = [1,3,5,7,9,11,13,15,17,19,21,23,25,27,29,31,33,35,37,39,41,43,45,47,49,51,53,55,57,59,61,63,65,67,69,71,73,75,77,79,81,83,85,87,89,91,93,95 = fixation;

2,4,6,8,10,12,14,16,18,20,22,24,26,28,30,32,34,36,38,40,42,44,46,48,50,52,54,56,58,60,62,64,66,68,70,72,74,76,78,80,82,84,86,88,90,92,94,96 = noreplace (circleAvoid\_ManikinTop,circleAvoid\_ManikinBottom,squareApproach\_ManikinTop, squareApproach\_ManikinBottom)]

</block>

<block approach\_ap>

/ onblockbegin = [values.test = 1;]

/ screencolor = white

/ preinstructions = (approach\_ap)

/ trials = [1,3,5,7,9,11,13,15,17,19,21,23,25,27,29,31,33,35,37,39,41,43,45,47,49,51,53,55,57,59,61,63,65,67,69,71,73,75,77,79,81,83,85,87,89,91,93,95 = fixation;

2,4,6,8,10,12,14,16,18,20,22,24,26,28,30,32,34,36,38,40,42,44,46,48,50,52,54,56,58,60,62,64,66,68,70,72,74,76,78,80,82,84,86,88,90,92,94,96 = noreplace (ApApproach\_ManikinTop,ApApproach\_ManikinBottom,SedenAvoid\_ManikinTop,SedenAvoid\_ManikinBottom)]

</block>

<block avoid\_ap>

/ onblockbegin = [values.test = 1;]

/ screencolor = white

/ preinstructions = (avoid\_ap)

/ trials = [1,3,5,7,9,11,13,15,17,19,21,23,25,27,29,31,33,35,37,39,41,43,45,47,49,51,53,55,57,59,61,63,65,67,69,71,73,75,77,79,81,83,85,87,89,91,93,95 = fixation;

2,4,6,8,10,12,14,16,18,20,22,24,26,28,30,32,34,36,38,40,42,44,46,48,50,52,54,56,58,60,62,64,66,68,70,72,74,76,78,80,82,84,86,88,90,92,94,96 = noreplace (ApAvoid\_ManikinTop,ApAvoid\_ManikinBottom,SedenApproach\_ManikinTop,SedenApproach\_ManikinBottom)]

</block>

<block images>

/ trials = [1= instructionimages]

</block>

<block reminder>

/ trials = [1=reminder]

</block>

<block Consent>

/trials = [1 = Consent]

/ screencolor = (white)

</block>

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Informed consent

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

<html Consent>

/items = ("Consent\_Form\_EN.html")

/position = (50%, 40%)

/size = (70%, 70%)

</html>

<trial Consent>

/inputdevice = mouse

/stimulusframes = [1 = Consent, agree, disagree]

/validresponse = (agree, disagree)

/monkeyresponse = ("agree")

/ correctresponse = (agree)

/ontrialend = [if (trial.Consent.response == "disagree") script.abort()]

</trial>

<text agree>

/items = ("Yes, ~nI want to participate.")

/position = (25%, 85%)

/ fontstyle = ("Arial", 2%, false, false, false, false, 5, 1)

/ txcolor = (white)

/ txbgcolor = (black)

/ size = (20%, 10%)

/ vjustify = center

</text>

<text disagree>

/items = ("No, ~nI do not want to participate.")

/position = (75%, 85%)

/ fontstyle = ("Arial", 2%, false, false, false, false, 5, 1)

/ txcolor = (white)

/ txbgcolor = (black)

/ size = (20%, 10%)

/vjustify = center

</text>

\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Dermographics

\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\*\* participant \*\*\*

<textbox participant>

/ caption = "Please enter your participant code below, organized as follows:

Please write in capital letters:

1. The first two letters of your first parent's first name (e.g., DA)

2. The first two letters of your second parent’s first name (e.g., NI)

3. The last two letters or numbers of your postal code (e.g.,K5)

4. The last letter of your last name (e.g., M)

5. The letters E and N corresponding to the language in which you are completing the study (English, e.g., EN)"

/ textboxsize = (80, 15)

/ mask = alphanumeric

/ required = true

/ fontstyle = ("Arial", 2.5%, false, false, false, false, 5, 1)

</textbox>

<surveypage participant>

/ caption = " "

/ fontstyle = ("Arial", 2.5%, true, false, false, false, 5, 1)

/ questions = [1=participant]

/ finishlabel = "Next"

/ showpagenumbers = false

/ nextlabel = "Next"

/ backlabel = " "

</surveypage>

<block participant>

/ trials = [1=participant]

</block>

\*\*\* Demographics \*\*\*

<textbox age>

/ caption = "What is your age (in years)?"

/ textboxsize = (80, 15)

/ mask = alphanumeric

/ fontstyle = ("Arial", 2.5%, false, false, false, false, 5, 1)

/ subcaption = "Please answer this question with only numbers."

/ subcaptionfontstyle = ("Arial", 2.5%, false, true, false, false, 5, 1)

/ required = true

</textbox>

<radiobuttons sex>

/ caption = "What is your sex?"

/ options = ("Male", "Female")

/ fontstyle = ("Arial", 2.5%, false, false, false, false, 5, 1)

/ required = true

</radiobuttons>

<radiobuttons gender>

/ caption = "What is your gender identity?"

/ options = ("Man", "Woman", "Non binary", "Transgender man", "Transgender woman", "Other", "I prefer not to answer")

/ fontstyle = ("Arial", 2.5%, false, false, false, false, 5, 1)

/ required = true

</radiobuttons>

<textbox height>

/ caption = "What is your height in cm (e.g., for 1m79, please indicate 179)"

/ textboxsize = (80, 15)

/ mask = positiveinteger

/ fontstyle = ("Arial", 2.5%, false, false, false, false, 5, 1)

/ subcaption = "Please answer with numbers only."

/ subcaptionfontstyle = ("Arial", 2.5%, false, true, false, false, 5, 1)

/ required = true

</textbox>

<textbox weight>

/ caption = "What is your weight in kg? If you need to convert your weight in kg from lb please do lb/2.2"

/ textboxsize = (80, 15)

/ mask = positiveinteger

/ fontstyle = ("Arial", 2.5%, false, false, false, false, 5, 1)

/ subcaption = "Please answer this question with numbers only."

/ subcaptionfontstyle = ("Arial", 2.5%, false, true, false, false, 5, 1)

/ required = true

</textbox>

<textbox country>

/ caption = "What is your country of residence?"

/ textboxsize = (80, 15)

/ mask = alphanumeric

/ fontstyle = ("Arial", 2.5%, false, false, false, false, 5, 1)

/ subcaptionfontstyle = ("Arial", 2.5%, false, true, false, false, 5, 1)

/ required = false

</textbox>

\*Chronic health condition

<checkboxes chronic>

/ caption = "Has a doctor ever told you that you had any of the following conditions? If yes, please selected the conditions. If not, please select 'None'"

/ options = ("A heart attack including myocardial infarction or coronary thrombosis or any other heart problem including congestive heart failure",

"A stroke or cerebral vascular disease",

"High blood pressure or hypertension",

"High blood cholesterol",

"Diabetes or high blood sugar",

"Chronic lung disease such as chronic bronchitis or emphysema",

"Asthma",

"Arthritis, including osteoarthritis, or rheumatism",

"Osteoporosis",

"Cancer or malignant tumour, including leukaemia or lymphoma, but excluding minor skin cancers",

"Stomach or duodental ulcer, peptic ulcer",

"Parkinson disease",

"Hip fracture or femoral fracture",

"Alzheimer’s disease, dementia, organic brain syndrome, senility or any other serious memory impairment",

"Other affective or emotional disorders, including anxiety, nervous or psychiatric problems",

"Rheumatoid Arthritis",

"Chronic kidney disease",

"Other conditions, not yet mentioned",

"None")

/ fontstyle = ("Arial", 2.5%, false, false, false, false, 5, 1)

/ required = true

</checkboxes>

<surveypage demographics>

/ fontstyle = ("Arial", 25, true, false, false, false, 5, 1)

/ questions = [1=age; 2=sex; 3=gender; 4=height; 5=weight; 6=country; 7=chronic]

/ finishlabel = "Next"

/showpagenumbers = false

/ nextlabel = "Next"

/backlabel = " "

/ subcaption = ""

/ subcaptionfontstyle = ("Arial", 2.8%, false, false, false, false, 5, 1)

</surveypage>

<block demographics>

/ trials = [1=demographics]

</block>

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Questionnaires

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

<block survey>

/ trials = [1=IPAQ]

</block>

\*IPAQ

<surveypage IPAQ>

/ caption = "We are interested in finding out about the kinds of physical activities that people do as part of their everyday lives.

The questions will ask you about the amount of time you spent being physically active in the last 7 days.

Please answer each question even if you do not consider yourself to be an active person.

Please think about the the NUMBER OF DAYS and the TIME SPENT EACH OF THESE DAYS doing activities you do at work, as part of your house, and yard work,

to get from place to place, and in your spare time for recreation, exercise or sport."

/ fontstyle = ("Arial", 2.5%, false, false, false, false, 5, 1)

/ questions = [1=vigorous\_d; 2=vigorous\_m; 3=moderate\_d; 4=moderate\_m; 5=sedentary\_d; 6=sedentary\_m; 7=intention; 8=attitude1; 9=attitude2; 10=pain]

/ finishlabel = "Next"

/ showpagenumbers = false

/ nextlabel = "Next"

/ backlabel = " "

/ subcaptionfontstyle = ("Arial", 2.8%, false, false, false, false, 5, 1)

</surveypage>

<textbox vigorous\_d>

/ caption = "Think about all the VIGOROUS activities that you did in the LAST 7 DAYS. Vigorous physical activities refer to activities that take hard physical effort and make you breathe much harder than normal.

During the last 7 days, on how many days did you do vigorous physical activities like heavy lifting, digging, aerobics, or fast bicycling?

Number of days per week:"

/ textboxsize = (10, 5)

/ fontstyle = ("Arial", 2.5%, false, false, false, false, 5, 1)

/ mask = alphanumeric

/ required = true

</textbox>

<textbox vigorous\_m>

/ caption = "Number of minutes per day:"

/ textboxsize = (10, 5)

/ fontstyle = ("Arial", 2.5%, false, false, false, false, 5, 1)

/ mask = alphanumeric

/ required = true

</textbox>

<textbox moderate\_d>

/ caption = "Think about all the MODERATE activities that you did in the LAST 7 DAYS. Moderate activities refer to activities that take moderate physical effort and make you breathe somewhat harder than normal.

During the last 7 days, on how many days did you do moderate physical activities like carrying light loads, bicycling at a regular pace, or doubles tennis? Do not include walking.

Number of days per week: "

/ textboxsize = (10, 5)

/ fontstyle = ("Arial", 2.5%, false, false, false, false, 5, 1)

/ mask = alphanumeric

/ required = true

</textbox>

</textbox>

<textbox moderate\_m>

/ caption = "Number of minutes per day:"

/ textboxsize = (10, 5)

/ fontstyle = ("Arial", 2.5%, false, false, false, false, 5, 1)

/ mask = alphanumeric

/ required = true

</textbox>

<textbox sedentary\_d>

/ caption = "The last question is about the time you spent SITTING during the LAST 7 DAYS. Include time spent at work, at home, while doing course work and during leisure time.

This may include time spent sitting at a desk, visiting friends, reading, or sitting or lying down to watch television.

During the last 7 days, how much time did you spend sitting?

Number of days per week: "

/ textboxsize = (10, 5)

/ fontstyle = ("Arial", 2.5%, false, false, false, false, 5, 1)

/ mask = alphanumeric

/ required = true

</textbox>

<textbox sedentary\_m>

/ caption = "Number of minutes per day:"

/ textboxsize = (10, 5)

/ fontstyle = ("Arial", 2.5%, false, false, false, false, 5, 1)

/ mask = alphanumeric

/ required = true

</textbox>

\* Intention

<radiobuttons intention>

/ caption = "How much do you agree with following statement:

Over the next 7 days,

I intend to do at least 150 minutes of moderate-intensity physical activity;

or at least 75 minutes of vigorous intensity physical activity;

or an equivalent combination of moderate- and vigorous-intensity physical activity."

/ options = ("1~nStrongly ~nDisagree","2~n~n","3~n~n","4~n~n","5~n~n","6~n~n","7~nStrongly ~nAgree")

/ optionvalues = ("1", "2", "3", "4", "5", "6", "7")

/ fontstyle = ("Arial", 2.5%, false, false, false, false, 5, 1)

/ orientation = horizontalequal

/ required = true

</radiobuttons>

\* Explicit affective attitude

<radiobuttons attitude1>

/ caption = "For me, to participate in regular physical activity is ..."

/ options = ("1~nUnpleasant","2~n~n","3~n~n","4~n~n","5~n~n","6~n~n","7~nPleasant")

/ optionvalues = ("1", "2", "3", "4", "5", "6", "7")

/ fontstyle = ("Arial", 2.5%, false, false, false, false, 5, 1)

/ orientation = horizontalequal

/ required = true

</radiobuttons>

<radiobuttons attitude2>

/ caption = "For me, to participate in regular physical activity is ..."

/ options = ("1~nUnenjoyable","2~n~n","3~n~n","4~n~n","5~n~n","6~n~n","7~nEnjoyable")

/ optionvalues = ("1", "2", "3", "4", "5", "6", "7")

/ fontstyle = ("Arial", 2.5%, false, false, false, false, 5, 1)

/ orientation = horizontalequal

/ required = true

</radiobuttons>

\*\*\*\*\*\*\*\*\*\*\*\*\*

Randomization

\*\*\*\*\*\*\*\*\*\*\*\*\*

<expt>

/subjects = (1 of 4)

/groupassignment = random

/ groups = (1 of 4)

/ blocks = [

1 = Consent;

2 = participant;

3 = demographics;

4 = approach\_circle;

5 = reminder;

6 = images;

7 = approach\_ap;

8 = reminder;

9 = images;

10 = avoid\_ap;

11 = reminder;

12 = avoid\_circle;

13 = survey

]

/ preinstructions = (intro)

/ postinstructions = (end)

/ onexptend = [values.completed = 1]

</expt>

<expt>

/subjects = (2 of 4)

/ groups = (2 of 4)

/groupassignment = random

/ blocks = [

1 = Consent;

2 = participant;

3 = demographics;

4 = avoid\_circle;

5 = reminder;

6 = images;

7 = avoid\_ap;

8 = reminder;

9 = images;

10 = approach\_ap;

11 = reminder;

12 = approach\_circle;

13 = survey

]

/ preinstructions = (intro)

/ postinstructions = (end)

/ onexptend = [values.completed = 1]

</expt>

<expt>

/subjects = (3 of 4)

/groupassignment = random

/ groups = (3 of 4)

/ blocks = [

1 = Consent;

2 = participant;

3 = demographics;

4 = avoid\_circle;

5 = reminder;

6 = images;

7 = approach\_ap;

8 = reminder;

9 = images;

10 = avoid\_ap;

11 = reminder;

12 = approach\_circle;

13 = survey

]

/ preinstructions = (intro)

/ postinstructions = (end)

/ onexptend = [values.completed = 1]

</expt>

<expt>

/subjects = (4 of 4)

/ groups = (4 of 4)

/groupassignment = random

/ blocks = [

1 = Consent;

2 = participant;

3 = demographics;

4 = approach\_circle;

5 = reminder;

6 = images;

7 = avoid\_ap;

8 = reminder;

9 = images;

10 = approach\_ap;

11 = reminder;

12 = avoid\_circle;

13 = survey

]

/ preinstructions = (intro)

/ postinstructions = (end)

/ onexptend = [values.completed = 1]

</expt>