

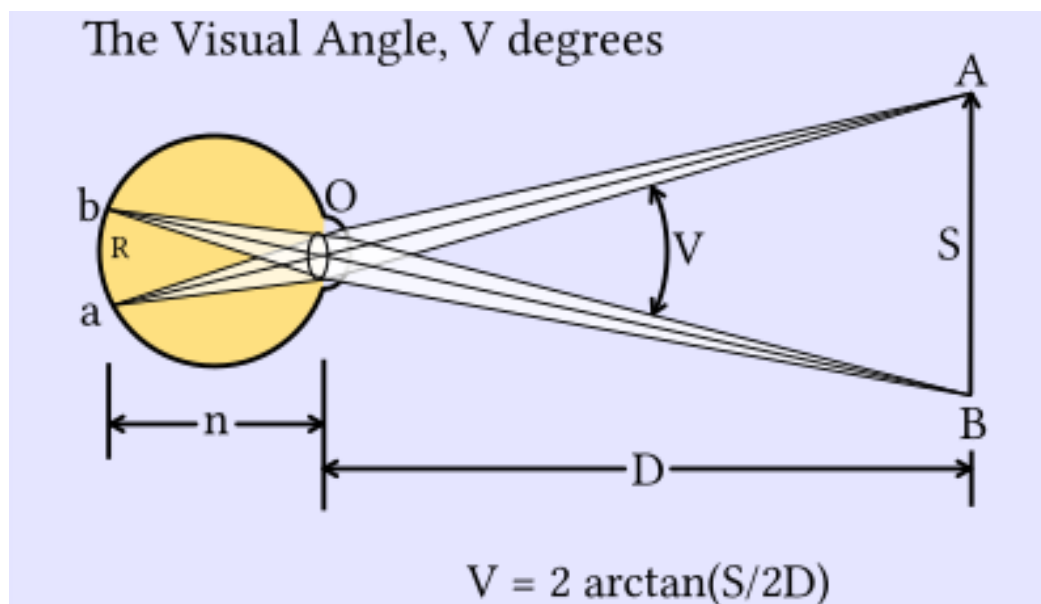
PRAKTIKUM 2: SILMALIIGUTUSED

Silmaliigutustest ning silmaandurite reliaablusest ja valiidsust rääkides kasutatakse tihti nägemisnurga kraade. Lisaks kasutatakse sama parameetrit erialases kirjanduses stiimulite suuruste väljendamiseks. Selles praktikumis õpimegi monitori sätteid muutma selliselt, et PsychoPy programm väljendaks stiimulite suurusi usaldusväärselt nägemisnurga kraadides või sentimeetrites. Loo lihtsa programmi Optokineetilise nüstagmi esilekutsumiseks, kus kasutame nägemispsühholoogias laialdaselt kasutatavat võrestiiimulit ja väljendame selle suurust nägemisnurga kraadides. Õpime tõlgendama ja muutma võrestiiimuli põhilisemaid parameetreid nagu ruumiline laotustihedus ja orientatsioon.

1. Nägemisnurga kraadide arvutamine

Ülesanne 1. Leia enda ettesirutatud pöidla jämedaima koha laius nägemisnurga kraadides. Nurgakraadide leidmiseks ja teiste tulemustega võrdlemiseks saad kasutada ka [seda Exceli arvutustabelit](#).

Visuaalsete stiimulite suurusi ja reetina omadusi kirjeldatakse nägemispsühholoogias tihti nurgakraadides, mille leidmise valem on toodud järgmisel joonisel, kus S vastab stiimuli küljepikkusele ja D objekti kaugusele silmast. Exceli arvutustabelis saab nurgakraadi leidmiseks kasutada järgmist süntaksit: $\text{DEGREES}(2 * \text{ATAN}(S / (2 * D)))$. Tihti raporteeritakse, et kuvari kaugus katseisiku silmast oli täpselt 57 cm. Kas oskad välja nuputada miks?¹



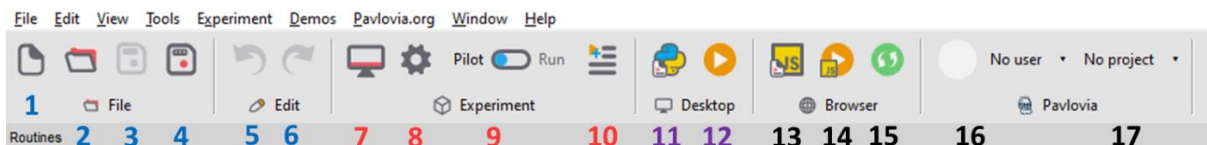
¹ Vihje: leia järgnevate küljepikkustega objektide pikkused nägemisnurga kraadides (1, 3, 5 cm), kui objekt paikneb 57 cm kaugusel.

2. Monitori seadistamine

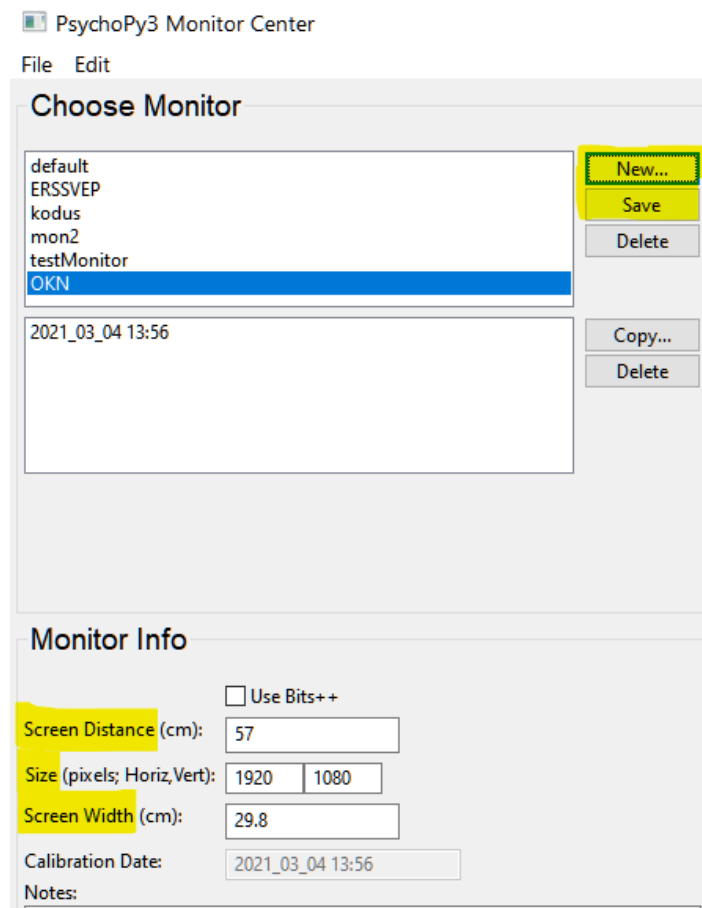
Ülesanne 2. Loo uus monitori seadete fail ja anna sellele nimeks OKN (Optokineetiline nüstagm) ja muuda selle järgmisi parameetreid²:

- ekraani kaugus katseisikust sentimeetrites: 57
- ekraani suurus pikslites: 1920 x 1080
- ekraani laius sentimeetrites: 29.8

Selleks, et saaksime stiimulite suuruseid kraadides või näiteks sentimeetrites väljendada tuleks muuta PsychoPy monitori seadeid. Seadete muutmiseks klõpsa ikoonide menüüs monitori pildiga ikoonil (7).



Ikoonide menüü



PsychoPy monitori seaded

² Need on ühele sülearvutile vastavad näiteparameetrid, mida tuleks vastavalt katses kasutatavatele seadmetele ja mõõtmistingimustele kohandada.

Monitori pikslitiheduse väljaselgitamine

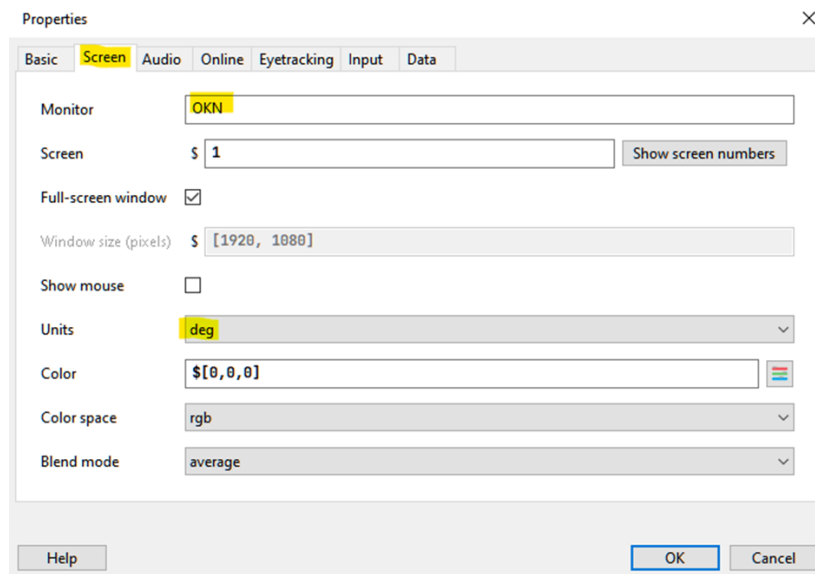
Arvuti monitori resolutsiooni väljaselgitamiseks mine enda operatsioonisüsteemi monitori seadetes, kus saadki kontrolli millist resolutsiooni monitor hetkel kasutab.

- Windowsi arvutil: Desktopil parem hiireklõps: [Display settings](#) > [Display resolution](#) (nt 1920 x 1080)
- Macintoshi arvutil: Vasak hiireklõps Apple'i ikoonil: [System Preferences...](#) > [Displays](#) > [Resolution](#) > [Scaled](#) (kui teha linnuke Scaled ette, siis peaks näitama, milline resolutsioon hetkel aktiivne on)

Ülesanne 3. Muuda eksperimendi seadeid selliselt, et PsychoPy kasutaks stiimulite esitamisel eelmises ülesandes loodud monitori seadeid (OKN). Vali eksperimendis kuvatavate stiimulite vaike ühikuteks nägemisnurga kraadid.

Eksperimendi seadete muutmiseks vajutame nutruga ikoonil (8) ja avame ekraani seadete saki (*Screen*), milles kirjutame esimesse lahtrisse (*Monitor*) eelmises ülesandes salvestatud monitori seadete nime.

Vaikeühikute muutmiseks vali *Units* järel *deg*.



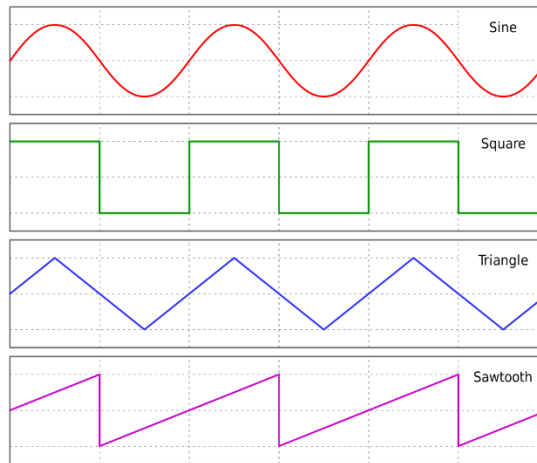
Eksperimendi seadete menüü.

3. Visuaalsete stiimulite baasparameetrid: orientatsioon ja ruumiline laotustihedus

Nägemispsühholoogia katsed nõuavad visuaalsete stiimulite baasparameetrite väga täpset kontrollimist. Värvuse, suuruse ja liikumise kõrval on sarnasteks omadusteks veel näiteks orientatsioon ja ruumiline laotustihedus. Alumisel joonisel on toodud näide, kus nägemiskatsetes tihti kasutatava hajutatud piirjoonega siinusvõre puhul ongi neid kaht dimensiooni (orientatsioon ja ruumiline laotustihedus) süstemaatiliselt varieeritud. Siinusvõre on visuaalne stiimul, mida iseloomustab ühtlaselt ja perioodiliselt muutuv valgusjõud.



Ülemisel paneelil (a) on varieeritud süstemaatiliselt siinusvõre orientatsiooni, kuid ruumiline laotustihedus on konstantne – alumisel paneelil (b) on tehtud aga täpselt vastupidi ehk süstemaatiliselt varieerub siinusvõre laotustihedus, aga orientatsioon ei muutu.



Näide neljast erinevast võre tüübist. Vertikaaltelg tähistab valgusjõudu ja horisontaaltelg annab edasi ruumilist mõõdet. Joonte tipud vastavad kõrgemale valgusjõule ehk valgetele aladele ja madalamad tumedamatele aladele. Oletame, et joonise horisontaaltelje kogupikkuseks on üks nägemisnurga kraad, siis sellisel juhul ütleme, et võre ruumiliseks laotustiheduseks on 3 Hz, sest ühe kraadi jooksul muutub valgusjõud perioodiliselt kolm korda.

Järgmiseks loome lihtsa programmi Optokineetilise nüstagmi esilekutsumiseks, milles kasutame siinusvõrede asemel diskreetse (järsu) üleminekuga võresid. Eelmine joonis demonstreerib valgusjõu muutumist erinevat liiki võredes, kus kõige ülemine punase joonega paneel (Sine) vastab siinusvõrele ja selle all paiknev roheline joonega paneel aga võretüübile, mida järgmisena loodavas programmis kasutama hakkame (Square).

Ülesanne 4. Vali komponentide menüüst võre stiimul (*Grating*), lisa see *trial* rutiinilemendi koosseisu ja muuda selle *Basic* saki alla olevaid parameetreid:

- a) algushetkeks 0 ja jäta kestuse lahter tühjaks

grating Properties

Basic Layout Appearance Texture Data Testing

Name: grating

time (s): 0.0

Expected start (s):

duration (s):

Expected duration (s):

Start Stop

Help OK Cancel

Võrestiiimuliga seotud parameetrite menüü (Basic).

- b) laiuseks 30 ja kõrguseks 20 nägemisnurga kraadi

grating Properties

Basic Layout Appearance Texture Data Testing

Size [w,h]: (30, 20)

Position [x,y]: (0, 0)

Spatial Units: from exp settings

Anchor: center

Orientation: 0

constant constant

Help OK Cancel

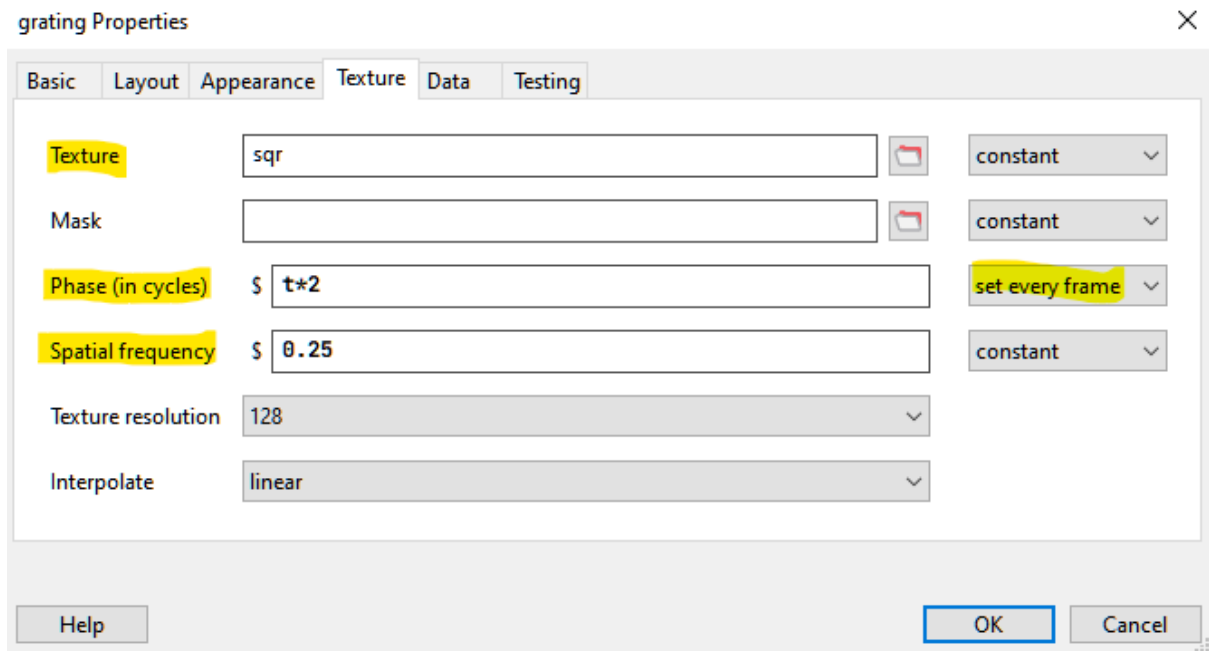
Võrestiiimuliga seotud parameetrite menüü (Layout).

- c) NB! Kuna oleme kasutatud suurusühikut keskselt ära muutnud, siis iga komponendi juures seda enam defineerima ei peagi. Piisab, kui jääda vaikeväärtuse ehk *Spatial units* lahtris valida *from exp settings*.

4. Stiimulite parameetrite muutmine ajas

Ülesanne 4.1. Muuda võrekomponendi *Texture* saki all järgmisi parameetreid:

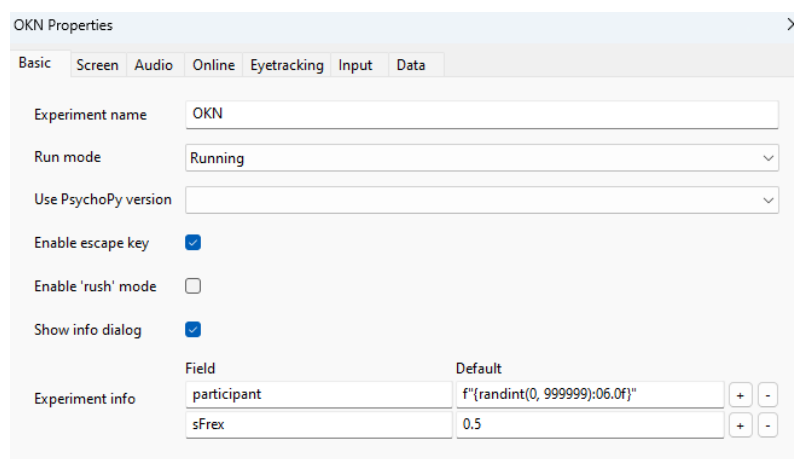
- kirjuta tekstuuri ribade ülemineku lahtrisse *sqr (square)*
- sea faas igal ekraani värskendusel muutuma sammuga $t \cdot 2$
- sea ruumiliseks sageduseks 2 kraadi



Võrestiiimuliga seotud parameetrite menüü (*Texture*).

Ülesanne 5. Muuda infokastikese sätteid selliselt, et programmi kasutaja saaks sinna sisestada ekraanile kuvatava võre ruumilise laotustiheduse. Anna uuele sisestuskastile nimetus *sFrex* ja määra selle vaikeväärtuseks 0.5 kraadi.

Infokastikese seadete muutmiseks klõpsa nutriga ikoonil (8). Avanenud seadete menüüs asenda rida *session* nimetusega *sFrex* (või mõne teise nimetusega, kuid hoia nimetus meeles, sest seda läheb meil kohe järgmises ülesandes tarvis) ja määra selle vaikeväärtuseks 2 kraadi. Soovi korral võib ka katseisiku (*participant*) aknasse sisestada vaikeväärtuse, kuid programmi funktsionaalsuse mõttes seda tarvis ei lähe (sinna sisestatud väärtus kirjutatakse hiljem andmetabelis *participant* veergu ja sama tekst ilmub ka andmetabeli nimetuse ette).



Eksperimendi sätted

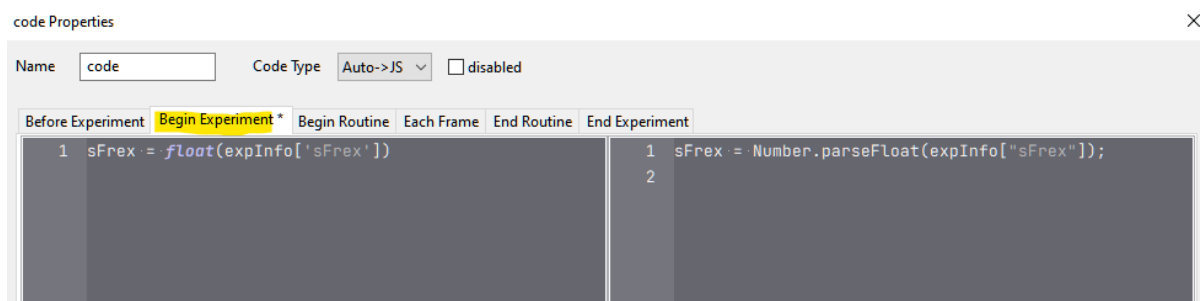
Ülesanne 5.1. Vali komponentide menüüst koodielement (*Custom* alaosast) ja lisa see *trial* rutiinielemendi koosseisu. Defineeri katse alguses uus muutuja *sFrex*, mis võtab rutiini

käivitamisel infokastikesest kasutaja sisestatud väärtuse ehk selle väärtuse, mille kasutaja sFreks järele lahtrisse kirjutas.

Ülesande lahendamiseks ava koodikomponendi seadetes katse algusele vastav sakk (*Begin Experiment*). Avanenud aknasse sisesta järgnev süntaks:

```
sFrex = float(explInfo['sFrex'])
```

sFrex on meie poolt defineeritud uus muutuja, mis saab enda väärtuse explInfo nime kandvast sõnastikust (ing *dictionary*). Sõnastikuks nimetatakse andmestruktuuri, mis hoiab endas võtmeid (infokasti read) ja neile vastavaid elemente (infokasti sisestatud tekst). Võtmeid ja väärtuseid võib olla erinevat tüüpi, kuid antud näites on mõlemad salvestatud sõnede ehk tekstina. Infokastidesse sisestatud andmete kättesaamiseks kirjutame sõnastiku järele kandiliste sulgude sisse jutumärkides tekstikasti nimetuse, mille sisse kasutaja soovitud ruumilise sageduse sisestas (sFrex).



Ülesanne 5.2. Muuda võrekomponendi *Advanced* saki all järgmisi parameetreid:

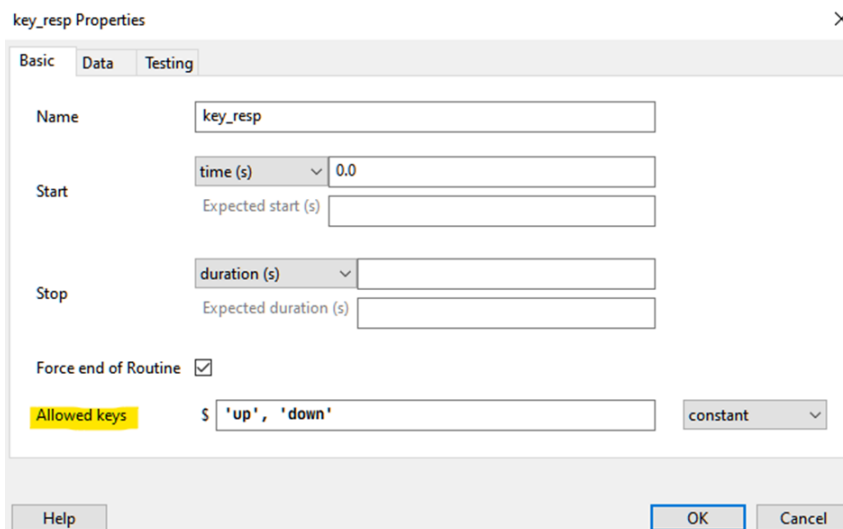
- asenda ülesandes 4.1 sisestatud ruumilise laotustiheduse väärtus eelmises ülesandes defineeritud muutujaga sFrex;

Ülesande lahendamiseks ava ülesandes 4.1 loodud võrekomponendi *Advanced* sätted ja kirjuta ruumilise laotustiheduse (Spatial frequency \$) aknasse float(sFrex), kus sFrex on meie poolt eelnevalt loodud muutuja.

Teine võimalus sama ülesande täitmiseks on explInfo['sFrex'] otse laotustiheduse lahtrisse kanda, kuid lisaülesannetega edasi liikumiseks on kasulikum kasutada eelpool kirjeldatud lahendust.

Lisaülesanded: klahvivajutuse kontrollimine koodielemendis

Ülesanne 6. Lisa trial rutiinielemeendile vastusekomponent. Määra vastuseklahvideks üles ja all („up“ ja „down“).

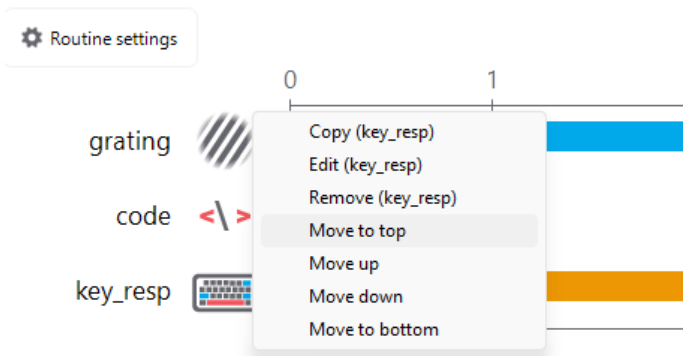


The 'key_resp Properties' dialog box has three tabs: Basic, Data, and Testing. The Basic tab is active. It contains the following fields and controls:

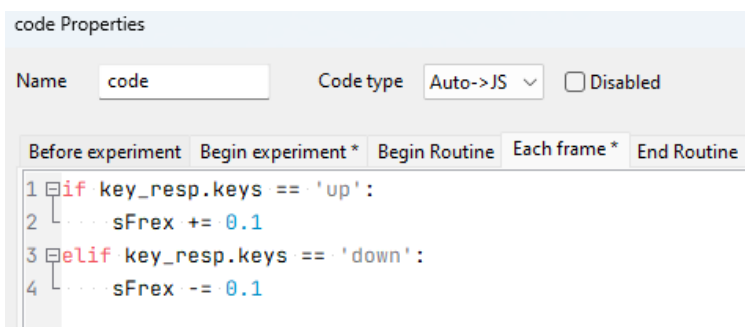
- Name:** A text field containing 'key_resp'.
- Start:** A dropdown menu set to 'time (s)' with a value of '0.0'. Below it is an 'Expected start (s)' text field.
- Stop:** A dropdown menu set to 'duration (s)' with an empty text field. Below it is an 'Expected duration (s)' text field.
- Force end of Routine:** A checked checkbox.
- Allowed keys:** A text field containing '\$ 'up', 'down'' and a dropdown menu set to 'constant'.

At the bottom are buttons for 'Help', 'OK', and 'Cancel'.

Liiguta vastuseklahvi komponent rutiinide menüüs kõige ülemisse asendisse:



Ülesanne 7. Lisa koodikomponendi *Each Frame* saki alla tingimuse, mis kontrolliks, kas kasutaja vajutas üles või alla klahvi ja muudaks vastavalt sellele ruumilist laotustihedust 0.1 kraadi võrra.³



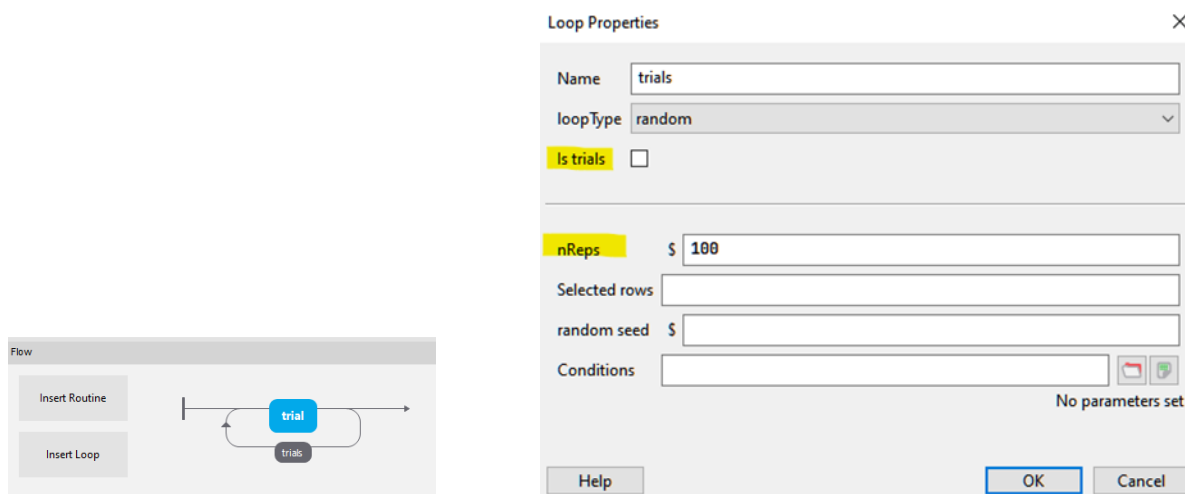
The 'code Properties' dialog box has tabs for 'Before experiment', 'Begin experiment *', 'Begin Routine', 'Each frame *', and 'End Routine'. The 'Each frame *' tab is active. It shows the following code:

```

1 if key_resp.keys == 'up':
2     ...sFrex += 0.1
3 elif key_resp.keys == 'down':
4     ...sFrex -= 0.1
  
```

³ Tingimusesete kasutamist Pythoni programmeerimiskeeles selgitab hästi veebis kättesaadav ja Tartu Ülikooli arvutiteaduse instituudi töörühma poolt koostatud programmeerimisõpik: <https://web.htk.tlu.ee/digitaru/programmeerimine/chapter/tingimused/>

Ülesanne 8. Lisa trial rutiinielemendile katseskeemi (*Flow*) menüüs aas ja võta selle sätetes *Is trials* järel linnuke ära. Määra korduste arvaks mingi suur arv (100), et programm kohe pärast paari vajutust kinni ei läheks.



Ülesanne 9. Lisa rutiinielemendile *trial* tekstikomponent, mis kuvab ekraani paremas nurgas ekraanile esitatud võre ruumilise laotustihedus. Muuda tekstikomponendi järgmisi parameetreid:

- muuda tekstikomponendi kestust selliselt, et komponent jääks rutiinielemendi lõpuni ekraanile;
- sea teksti suuruseks 1 kraad;
- sea tekstikomponendi x-telje asukohaks 18 ja y-telje asukohaks -7 kraadi keskkoha suhtes;
- anna teksti väärtuseks jooksva laotustiheduse väärtus ehk kuuendas ülesandes defineeritud **sF_{ref}** (soovi korral võid ka ümardada $\text{round}(\text{sF}_{\text{ref}}, 2)$)
- muuda teksti parameetreid selliselt, et selle väärtusi muudetaks pärast iga ekraanivärskendust.

Selle ülesande lahendamiseks vali komponentide menüüst tekstikomponent.

Tekstikomponendi rutiinielemendi lõpuni ekraanile jätmiseks jäta tekstielemendi kestust defineeriv kastike tühjaks. Teksti suuruse kraadides väljendamiseks tuleks kõigepealt *Advanced* sätete all muuta tekstielemendi ühikud (*Units*) kraadideks (*Deg*). Seejärel kirjuta tekstikasti kõrgust reguleerivasse lahtrisse (*Letter height*) number 1.

Kirjuta tekstikasti järgnev süntaks: $\$sF_{\text{ref}}$, kus dollarimärk annab PsychoPy-le teada, et tegemist pole tavalise tekstiga, vaid hoopis muutujaga.

text Properties

Basic Layout Appearance Formatting Data Testing

Name text

Start \$ time (s) 0.0
Expected start (s)

Stop \$ duration (s)
Expected duration (s)

\$round(sFrex,1)

Text

set every frame

Help OK Cancel

Tekstikomponendi seeded (Basic)

text Properties

Basic Layout Appearance Formatting Data Testing

Position [x,y] \$ (18, -7) constant

Spatial units from exp settings

Orientation \$ 0 constant

Tekstikomponendi seeded (Layout)

text Properties

Basic Layout Appearance Formatting Data Testing

Font Arial constant

Letter height \$ 1 constant

Language style LTR

Tekstikomponendi seeded (Formatting)