

Krossaspurningar

Spurning	Heiti spurningar	Stig	Spurningategund
i	Forsíða Viðmótsforritun Hlutapróf 2025		Upplýsingar eða hjálpargögn
i	2025 Inngangur að hluta 1		Upplýsingar eða hjálpargögn
1	Skrá fyrir notendaviðmót	5	Fjöl-val spurning
2	Java Module System	5	Fjöl-val spurning
3	FXML fx:id Spurning	5	Fjöl-val spurning
4	JavaFX Stage, Scene og Node	5	Fjöl-val spurning
5	JavaFX Layout Question	5	Fjöl-val spurning
6	include	5	Fjöl-val spurning
7	fx:define	5	Fjöl-val spurning
8	createString binding og vaktbreyta	5	Fjöl-val spurning
9	ListView toString	5	Fjöl-val spurning
10	Dialog - fxml	5	Fjöl-val spurning
11	Sérhæfur klasi notaður í fxml	5	Fjöl-val spurning
12	Dialog resultConverter	5	Fjöl-val spurning
13	JavaFX CSS og FXML	5	Fjöl-val spurning
14	MediaPlayer MediaView	5	Fjöl-val spurning

Opnar spurningar

Spurning	Heiti spurningar	Stig	Spurningategund
i	2025 Inngangur að hluta 2		Upplýsingar eða hjálpargögn

HlaupGanga

Spurning	Heiti spurningar	Stig	Spurningategund
15	HlaupGanga forritun	15	Forritun

Binding Listener

Spurning	Heiti spurningar	Stig	Spurningategund
16	BindingListener	15	Forritun

i Forsíða Viðmótsforritun Hlutapróf 2025

RAFRÆNT HLUTAPRÓF HBV201G VIÐMÓTSFORRITUN

Kennarar: Ebba Þóra Hvannberg (fyrirlestrar)

Deild: Iðnaðarverkfræði, vélaverkfræði og tölvunarfræðideild

Námskeið: HBV201G Viðmótsforritun

ETCS: 8

Vægi hlutapróf : 10% af lokaeyðingun en til hækkunar miðað við lokaprófseinkunn

Próftími: [8:20-9:40]

- **Prófreglur:** Prófið er tekið í Inspira prófakerfinu á tölvur nemenda sem eru stilltar þannig þeim er læst með Safe Exam Browser (SEB).
- **Hjálpargögn:** Engin hjálpargögn
- **Fylgigögn:** Athugið sérstaklega fylgigögn (resources) sem fylgja með prófinu.
- **Fyrirkomulag:** Prófið hefst á krossaspurningum. Þar eru nokkrir svarkostir taldir upp og nemandi velur „réttan eða réttasta“ svarmöguleikann hverju sinni. Ef nemandi á að velja fleiri en einn valkost er það tekið fram. Ekki er dregið niður fyrir röng svör. Í seinni hluta prófsins eru opnar spurningar.
- **Próftaka:** Mikilvægt er að lesa vel spurninguna áður en lokasvar er valið. Eftir að spurningu hefur verið svarað er ýtt á „áfram“ hnappinn (e. *Next*) eða næsta spurning valin á númeruðu flæðilínunni neðst á skjánum. Við aðgerðina vistast svarið sjálfkrafa en einnig vistast vinna nemanda sjálfkrafa á 15-20 sekúndna fresti. Mögulegt er að fara fram og til baka í prófinu og merkja við spurningar sem á eftir að svara.
- **Námsmat:** Einkunn fyrir þetta próf reiknast sjálfkrafa út frá krossaspurningum. Farið er yfir opnar spurningar samkvæmt matskvarða.

Gangi þér vel.

Ebba Þóra Hvannberg

i 2025 Inngangur að hluta 1

Hluti I. Fjölvallsspurningar. Samtals 70%. Opnu spurningarnar í hluta II gilda 30%

Í fjölvallsspurningunum kemur aðeins eitt svar til greina. Ekki er dregið frá fyrir rangt svar.

Ef þið sjáið ekki hluta prófsins, t.d. myndir eða fylgiskjöl, vinsamlegast hafið samband við prófvörð.

Ef galli á prófinu kemur í ljós, verður það skoðað við yfirferð prófsins. Ef nemandi er óviss um hvort skilningur hans á spurningu er réttur, getur hann tilgreint hver sá skilningur er í svarreit við opnum spurningum og/eða sent tölvupóst til kennara eftir prófið.

Eftirfarandi fylgigögn eru með

1 <https://prof.snara.is/>

2 <https://www.bing.com/translator>

3 <https://openjfx.io/javadoc/21/>

4 <https://openjfx.io/javadoc/21/javafx.graphics/javafx/scene/doc-files/cssref.html#typenumber>

5 vocabulary - PDF skjal

6 Java cheatsheet

7 HlaupGanga.java - fyrir dæmi 15

1 Skrá fyrir notendaviðmót

Í hvaða skrá er lýsing notendaviðmóts geymd?

Veldu einn valkost:

- ☐ UI.xml
- ☐ layout-view.fxml
- ☐ Layout.class
- ☐ styles.css
- ☐ MainApplication.java

Hámarksstig: 5

2 Java Module System

```
module hi.verkefni.vidmot{
    requires javafx.controls;
    requires javafx.fxml;
    opens hi.verkefni.vidmot to javafx.fxml;
    exports hi.verkefni.vidmot to javafx.graphics;
```

Hvað gerir

requires javafx.controls;

í eftirfarandi module-info.java skrá?

- ☐ Það leyfir öðrum módúlum að nota `javafx.controls` úr þessum módúl.
- ☐ Það gefur leyfi fyrir `javafx.controls` að nota `hi.verkefni.vidmot`.
- ☐ Það gefur til kynna að þessi módúll þarfnist `javafx.controls` til að keyra.
- ☐ Það takmarkar notkun `javafx.controls` við aðeins ákveðna pakka innan módúlsins.
- ☐ Það opnar pakkann `hi.verkefni.vidmot` fyrir `javafx.controls`.

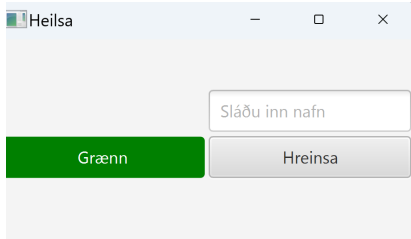
Hámarksstig: 5

3 FXML fx:id Spurning

```
<Button fx:id="jLitur" onAction="#skiptaUmLitHandler" prefHeight="29.0" prefWidth="167.0" style="-fx-background-col
<Label fx:id="jHalloNafn" prefHeight="48.0" prefWidth="154.0" GridPane.columnIndex="1" />
<TextField fx:id="jNafn" onAction="#heilsaHandler" prefHeight="30.0" prefWidth="156.0" promptText="Sláðu inn nafn" G
  <opaqueInsets>
    <Insets />
  </opaqueInsets>
</TextField>
<Button mnemonicParsing="false" onAction="#hreinsaHandler" prefHeight="30.0" prefWidth="180.0" text="Hreinsa" GridPa
```

Til fróðleiks er hér forrit í keyrslu

Að ofan er gefinn hluti úr FXML lýsingu á notendaviðmóti fyrir gluggann.



Í Controller klasanum eru tilviksbreytur fyrir þrjá viðmótshluti: **jLitur**, **jHalloNafn** og **jNafn**. Hins vegar hefur fjórði viðmótshlutinn (**Hreinsa** takki) ekki fx:id, og það er engin tilviksbreyta fyrir hann í Controller klasanum. Hvaða ástæða getur legið að baki?

Veldu réttasta svarið:

- ☐ Viðmótshluturinn er aðeins notaður í onAction handler (hreinsaHandler), svo það er ekki þörf á tilviksbreytu fyrir hann í Controller klasanum
- ☐ Hann hefur ekki fx:id, þannig að viðmótshluturinn verður ekki hluti af viðmótstrénu
- ☐ Tilviksbreytur eru aðeins nauðsynlegar fyrir viðmótshluti sem eru settir í VBox eða GridPane.
- ☐ Viðmótshluturinn er ekki sýnilegur í forritinu, svo það þarf ekki tilviksbreytu fyrir hann.
- ☐ FXML styður ekki að skilgreina hnappa án fx:id.

Hámarksstig: 5

4 JavaFX Stage, Scene og Node

Hvað lýsir best hlutverkum Stage, Scene og Node í JavaFX?

Veldu réttasta svarið

- ☐ Stage er gluggi forritsins, Scene inniheldur grafíska hluti, og Node er grunnbyggingareining sem er rót í erfðatré grafískra klasa í JavaFX.
- ☐ Stage er grafískur hlutur, Scene er gluggi forritsins, og Node er safn af öllu efni í viðmótinu.
- ☐ Stage stjórnar öllum viðmótshlutum, Scene býr til nýja glugga, og Node er aðeins notað fyrir texta.
- ☐ Stage er rót allra viðmótshluta, Scene er notað fyrir myndvinnslu, og Node táknar hljóðbrellur í forritinu.
- ☐ Stage, Scene og Node eru aðeins notaðir í JavaFX til að vinna með gagnagrunnstengingar.

Hámarksstig: 5

5 JavaFX Layout Question

Hvert eftirfarandi er **ekki** dæmi um layout (raðstjóra) í JavaFX?

Veldu réttasta svarið

- ☐ VBox
- ☐ GridPane
- ☐ BorderPane
- ☐ FlowManager
- ☐ StackPane

Hámarksstig: 5

6 include

```
<VBox xmlns="http://javafx.com/javafx/22"
      xmlns:fx="http://javafx.com/fxml/1">
  <fx:include source="Valmyndir.fxml"/>
</VBox>
```

Skoðaðu bútinn hér að ofan. Hver af eftirfarandi fullyrðingum er rétt?

Veldu eitt af eftirtöldu

- ☐ <fx:include> setur inn innihald annarrar .fxml skrár á sama stað og <fx:include> er notað.
- ☐ <fx:include> er aðeins notað fyrir CSS skrár og hefur engin áhrif á .fxml skrár.
- ☐ <fx:include> virkar eins og <fx:controller>, en leyfir að tengja margar .fxml skrár við sama stýriklasa (controller).
- ☐ <fx:include> er óþarft ef .fxml skráin inniheldur fx:root.
- ☐ <fx:include> býr sjálfkrafa nýjan Stage þegar skráin er hlaðin inn.

Hámarksstig: 5

7 fx:define

```
<VBox xmlns="http://javafx.com/javafx/22"
      xmlns:fx="http://javafx.com/fxml/1">
  <fx:define>
    <Label fx:id="myLabel" text="Halló heimur!" />
  </fx:define>
</VBox>
```

Skoðaðu ofangreint. Hvert eftirfarandi er rétt:

Veldu eitt af eftirtöldu

- ☐ <fx:define> er notað til að skilgreina hluti sem eru ekki sjálfkrafa afhjúpaðir í JavaFX viðmótstrénu.
- ☐ <fx:define> er notað til að stilla stjórnanda fyrir .fxml skrá.
- ☐ <fx:define> bætir sjálfkrafa öllum skilgreindum hlutum við JavaFX senuna (Sena).
- ☐ <fx:define> er aðeins notað til að skilgreina CSS stíl fyrir .fxml skrá.
- ☐ <fx:define> verður alltaf að vera fyrsta merkið í .fxml skrá til að það virki rétt.

Hámarksstig: 5

8 createString binding og vaktbreyta

```
<VBox spacing="10" xmlns="http://javafx.com/javafx/22" xmlns:fx="http://javafx.com/fxml/1" fx:controller="hi.property
  <Label fx:id="fxTími"/>
  <Button fx:id="fxHnappur" text="Uppfæra tíma" onAction="#uppfaeraTima"/>
</VBox>
```

```
@FXML private Label fxTími;
@FXML private Button fxHnappur;
private final ObjectProperty<LocalTime> valinnTími =
    new SimpleObjectProperty<>(LocalTime.now());
private final DateTimeFormatter formatter =
    DateTimeFormatter.ofPattern("HH:mm:ss");

@FXML
public void initialize() {
    fxTími.textProperty().bind(Bindings.createStringBinding(
        () -> valinnTími.get().format(formatter),
        valinnTími
    ));
}
@FXML
private void uppfaeraTima(ActionEvent actionEvent) {
    valinnTími.set(LocalTime.now());
}
```

Hér að ofan er FXML skrá og Java kóði sem sýnir hvernig binding og lambda fall eru notuð í JavaFX. Hvaða vaktbreyta (**observable**) er notuð í bindingunni í `fxTími.textProperty().bind(...)`?

Veldu einn valkost:

- ☐ `fxHnappur.setOnAction(e -> valinnTími.set(LocalTime.now()))`
- ☐ `valinnTími`
- ☐ `valinnTími.get().format(formatter)`
- ☐ `fxTími.textProperty`
- ☐ `formatter`

Hámarksstig: 5

9 ListView toString

```
@FXML
private ListView<Vedurstod> vedurstodvarView;
```

Skoðið eftirfarandi skilgreiningu

Til að nafn veðurstöðvarinnar birtist í listanum í viðmóttinu þyrfti að forrita eftirfarandi:

Veldu einn valkost:

- ☐ Kalla á `vedurstodvarView.toString()` í hvert skipti sem innihald listans er uppfært
- ☐ Ekkert til viðbótar
- ☐ Útfæra aðferðina `@Override public String toString() { return getName();}` í klasanum `Vedurstod`
- ☐ Kalla á `vedurstodvarView.toString()` í `initialize()` í `controller`num
- ☐ Útfæra aðferðina `getVedurstod()` í `Vedurstod`

Hámarksstig: 5

10 Dialog - fxml

Forritari hannar notendaviðmót fyrir dialog glugga sem inniheldur upplýsingar úr vinnslu-
klasanum **Vidskiptavinur**. Forritarinn lýsir dialognum í .fxml skrá. Hvaða klasi er í rótinni á
.fxml skránni?

Veldu einn valkost:

- ☐ DialogPane eða klasi sem erfir frá DialogPane
- ☐ Scene
- ☐ Dialog<Vidskiptavinur> sem inniheldur DialogPane
- ☐ Klasi sem erfir frá Vidskiptavinur
- ☐ DialogPane

Hámarksstig: 5

11 Sérhæfður klasi notaður í fxml

Þegar setja á hlut af sérhæfðum klasa (Custom component) **A** í .fxml skrá er það gert á
eftirfarandi hátt

Veldu einn valkost:

- ☐ <fx:include class=A />
- ☐ <A />
- ☐ <fx:component />
- ☐ <fx:root type=A />
- ☐ <fx:object A />

Hámarksstig: 5

12 Dialog resultConverter

Ef Dialog **d** á að skila gögnum úr dialog fyrir gögn úr leik þar sem klasinn **Game** inniheldur
gögnin, þá myndi setning sem opnar modal-glugga vera svona:

Veldu einn valkost:

- ☐ Game game = d.show();
- ☐ Optional<Game> result = d.showAndWait();
- ☐ Dialog<Game> d = new GameDialog(); d.showModal();
- ☐ Game game = d.getResult();
- ☐ d.open(Game.class);

Hámarksstig: 5

13 JavaFX CSS og FXML

```
protected void onRautt() {
    fxTexti.getStyleClass().removeAll("texti-green");
    fxTexti.getStyleClass().add("texti-red");
}
```

Skoðaðu ofangreindan handler í Controller klasa:

Hvað er texti-green og hvar er það skilgreint?

- ☐ texti-green er CSS-klasi sem er skilgreindur í .css skrá og tengdur við JavaFX forritið með stylesheets eigindinu í FXML.
- ☐ texti-green er skilgreint í FXML-skránni og stjórnar útliti textans þar.
- ☐ texti-green er aðferð í Controller-klasanum sem breytir lit textans í grænan.
- ☐ texti-green er stilling í .css skrá sem er sett beint á style eigindið í FXML.
- ☐ texti-green er fasti í Controller-klasanum sem geymir streng með litakóðanum fyrir grænan texta.

Hámarksstig: 5

14 MediaPlayer MediaView

```
private void newMediaPlayer(String mediaURL) {
    mediaPlayer = new MediaPlayer(new Media(mediaURL));
    mediaPlayer.setAutoplay(true);
    fxMediaView.setMediaPlayer(mediaPlayer);
}
```

Skoðaðu eftirfarandi kóðabút og svaraðu

Hvað gerir `setMediaPlayer(MediaPlayer mediaPlayer)` aðferðin í `MediaView` klasanum í JavaFX?

Veldu einn valkost:

- ☐ Hún breytir hljóðstyrknum í MediaPlayer samkvæmt MediaView.
- ☐ Hún byrjar sjálfkrafa spilun myndbandsins þegar MediaView er sýnt.
- ☐ Hún tengir MediaPlayer við Slider til að stýra hvar í myndbandinu er spilað
- ☐ Hún bindur MediaPlayer við MediaView til að birta myndband eða hljóð.
- ☐ Hún leyfir notandanum að velja skrá úr tölvunni til að spila í MediaPlayer.

Hámarksstig: 5

i 2025 Inngangur að hluta 2**Hluti II**

Í þessum hluta eru 2 spurningar sem gilda samtals 30%.

Gefinn er java ritill til að forrita í. Ritillinn er með setninganúmerum og sýnir java highlights með lit. Ritillinn hjálpar ykkur með réttan inndrátt á línum ef þið notið slaufusviga. Ritillinn athugar ekki hvort forritið er málfræðilega rétt og hann þýðir hvorki né keyrir forritið. Notið bestu venjur við forritun. Það þarf ekki að setja inn import setningar

Ef þið sjáið ekki hluta prófsins, t.d. myndir eða fylgiskjöl, vinsamlegast hafið samband við prófvörð.

Ef galli á prófinu kemur í ljós, verður það skoðað við yfirferð prófsins. Ef nemandi er óviss um hvort skilningur hans á spurningu er réttur, getur hann tilgreint hver sá skilningur er í svarreit við opnum spurningum og/eða sent tölvupóst til kennara eftir prófið.

Þið eigið að forrita skráningarforrit fyrir hreyfingu, hlaup og göngu. Notandi setur inn í textasviðin (TextField) fjölda km hlaup og fjölda km göngu. Notandi setur **ekki** inn <enter> í lokin á textasviðunum. Þegar notandi ýtir á hnappinn **Skrá hreyfingu** birtist í textasviðið samtals fjöldi hlaupinna km, samtals fjöldi genginna km síðan forritið var ræst og samtals hlaup og ganga. Notið vinnsluklasann HlaupGanga.java sem er gefinn í resources #7

Eftirfarandi tafla sýnir stöðuna í notendaviðmótinu eftir tvær skráningar

Skráning	Hlaup	Ganga	Samtals hlaup	Samtals ganga	Samtals hlaup og ganga
1	10	5	10	5	15
2	1	2	11	7	18

Eftirfarandi mynd sýnir notendaviðmót eftir að forritið er keyrt upp og búið er að færa inn hlaup og göngu (Skráning 1 hér að ofan) en **ekki** búið að ýta á Skrá hreyfingu. Þegar ýtt er á **Skrá** er Samtals hlaup 10 og samtals ganga 5.

Skrá hreyfingu

Hlaup (km): 10

Ganga (km): 5

Skrá

Samtals hlaup: 0 km

Samtals ganga: 0 km

Samtals hlaup og ganga: 0 km

Eftirfarandi mynd sýnir notendaviðmótið eftir að búið er að skrá aðra hreyfingu (Skráning 2 hér að ofan) og ýta á Skrá hreyfingu

Skrá hreyfingu

Hlaup (km):

Ganga (km):

Skrá

Samtals hlaup: 11 km

Samtals ganga: 7 km

Samtals hlaup og ganga: 18 km

15 HlaupGanga forritun

```
public class HlaupGangaController {  
    // tilviksbreytur  
    public TextField fxHlaup;  
    public Label fxHlaupSamtals;  
  
    // forritið hér viðbótarbreytur fyrir viðmótshluti - gefið ykkur breytuheitin  
  
    // vinnsla - forritið  
  
    // aðferðir  
    /**  
     * Aðferð fyrir Skrá hreyfingu hnappinn  
     * @param actionEvent ónotað  
     */  
    public void skraHreyfingu(ActionEvent actionEvent) {  
        // forritið  
    }  
}
```

Hér er beinagrind af HlaupGangaController.java - afritið í ritilinn

Settu svarið þitt inn hér

1

Hámarksstig: 15

```
<TextField fx:id="efri" promptText="Sláðu inn texta hér"/>
<Label fx:id="label"/>
<TextField fx:id="nedri" promptText="Sláðu inn texta hér"/>
<Label fx:id="uppercaseLabel" />
```

Forrit hefur tvö TextField (**efri**, **nedri**) og einn Label hlut (**label**). Forritið hefur eftirfarandi virkni:

1. **label** hluturinn á að hafa það sama og er slegið inn í **efri**. Útfærið með **bind**

2. Þegar **nedri** hluturinn breytist setjið **nýja** textann í hástafi og setjið í **uppercaseLabel** og prentið út á console **breyttist + nýja gildið**. Útfærið með **addListener**

bútur úr .fxml skránni er hér að ofan

Visbending: SimpleStringProperty útfærir Property interface sem hefur aðferðina bind

```
void bind (ObservableValue<? extends T> observable)
```

og gæti byrjað svona ef x er hlutur með textProperty() aðferð

```
x.textProperty().bind( );
```

Visbending: String klasinn hefur tilviksaðferðina **toUpperCase** til að breyta streng í hástafi

```
public String toUpperCase ()
```

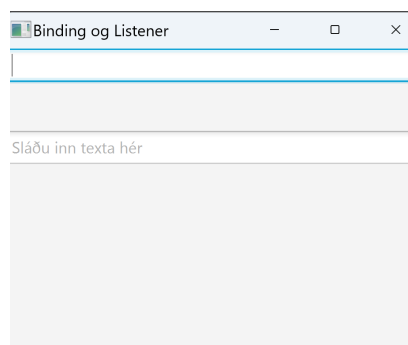
Visbending: Þið getið notað eftirfarandi beinagrind að notkun með **addListener** sem er aðferð á ObservableValue interface sem SimpleStringProperty útfærir, en textProperty() aðferðin skilar því. Ef x er hlutur með textProperty aðferð þá gætirðu byrjað með og breytt.

Visbending: Þú getur notað eftirfarandi skeleton. **addListener** er aðferð á ObservableValue interface sem SimpleStringProperty útfærir, en textProperty() skilar því. Tög breytu fyrir **addListener lambda** fallið eru ObservableValue<extends String>, String, String

Ef x er hlutur með textProperty aðferð þá gætirðu byrjað með og breytt.

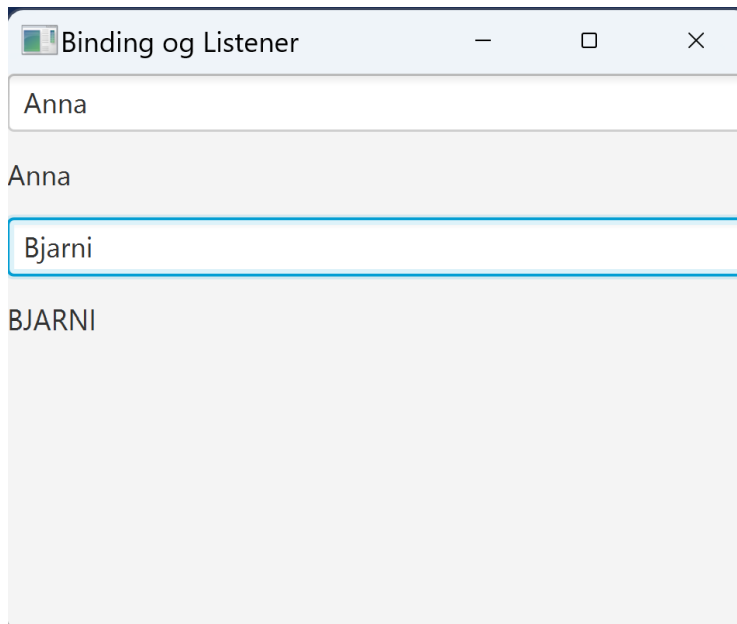
```
x.textProperty().addListener((vaktad, gamla, nýja) -> { });
```

Þegar forritið er ræst þá lítur viðmótið svona út



Þegar slegið hefur verið inn Anna í **efri** og Bjarni í **nedri** lítur viðmótið svona út og eftirfarandi er prentað á console

```
nedri breyttist: B
nedri breyttist: Bj
nedri breyttist: Bja
nedri breyttist: Bjar
nedri breyttist: Bjarn
nedri breyttist: Bjarni
```



Gefin er beinagrind að controller fyrir viðmótið. Ljúkið við að forrita.

16 BindingListener

```
public class BindingListenerController
{
    @FXML
    private TextField efri;
    @FXML
    private TextField nedri;
    @FXML
    private Label label;
    @FXML
    private Label uppercaseLabel;
    @FXML
    public void initialize() {
        // Binding: label hefur sama texta og efri
        // forritið hér - breytið
        x.textProperty().bind();

        // Þegar nedri hluturinn breytist þá er nýi textinn settur í hástafi og settur í uppercaseLabel
        // og prentað út á console "breyttist" + nýja gildið
        // forritið hér - breytið
        x.textProperty().addListener((vaktad, gamla, nyja) -> { });
    }
}
```

Fyrir ofan er beinagrind að controller. Afritaðu hann í ritilinn

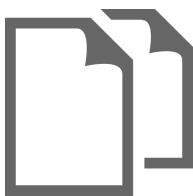
Settu svárið þitt inn hér

1	
---	--

Hámarksstig: 15

Document 1

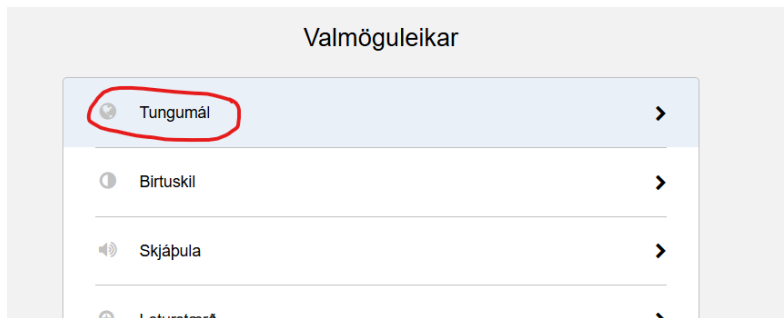
Attached



This exam is offered in Icelandic and English. To change to English please click on the ,hamburger' menu in the top right corner of the exam:



Then click on ,tungumál' or ,languages':



And select ,Enska' or ,English':

