

i Inngangur að Hluta I

Hluti I. Fjölvalsspurningar. Samtals 30%. Opnu spurningarnar í hluta II gilda 70%
Í fjölvalsspurningunum 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.

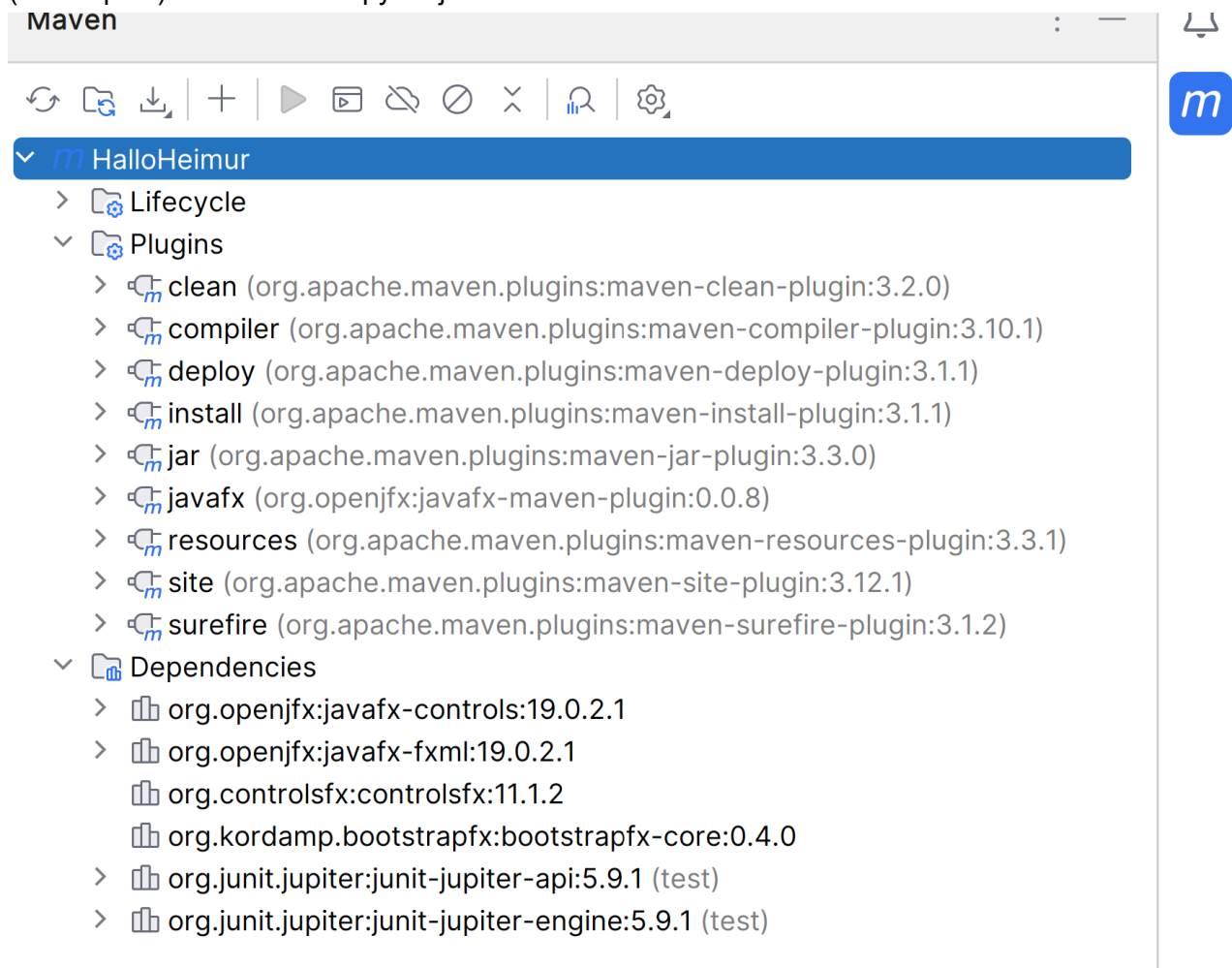
Ef galli á prófinu kemur í ljós, verður það skoðað við yfirferð prófsins. Ef nemandi er óviss um hvort skilningur hans á spurningu er réttur, getur hann tilgreint hver sá skilningur er í svarreit við opnum spurningum og/eða sent tölvupóst til kennara eftir prófið.

Eftirfarandi fylgigögn eru með

- 1 <https://prof.snara.is/>
- 2 <https://openjfx.io/javadoc/21/>
- 3 <https://openjfx.io/javadoc/21/javafx.graphics/javafx/scene/doc-files/cssref.html#typenumber>
- 4 vocabulary - PDF skjal
- 5 leiðbeiningar um útfyllingu skanblaða - PDF skjal
- 6 Scenario GoOut - PDF skjal
- 7 Lan.java - PDF skjal
- 8 Java cheatsheet

1 Maven gluggi

Eftirfarandi er mynd af Maven uppsetningu fyrir javaFX forrit. Hvaða útgáfa af java þýðandanum (e. compiler) er notuð til að þýða .java skrár í verkefninu?



Veldu eitt af eftirtöldu

- ☐ 21.0.1
- ☐ 19.0.2.1
- ☐ 3.10.1
- ☐ sama og er í JRE_HOME umhverfis breytunni (environment variable)
- ☐ Það sést ekki út frá þessum upplýsingum

Maximum marks: 2

2 Maven pom.xml property

Skoðið eftirfarandi setningar í **pom.xml** skrá. Hvaða setningu vantar í eftirfarandi kóða sem er merktur með commenti ?

```
<properties>
  <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
  <junit.version>5.9.1</junit.version>
  <!-- hvað vantar hér á eftir -->
</properties>
<dependency>
  <groupId>org.openjfx</groupId>
  <artifactId>javafx-controls</artifactId>
  <version>${javafx.version}</version>
</dependency>
```

Veldu eitt af eftirtöldu

- ☐ <maven.compiler.release>21</maven.compiler.release>
- ☐ org.openjfx:javafx-controls:19.0.2.1
- ☐ <version>19.0.2.1</version>
- ☐ <javafx.version>19.0.2.1</javafx.version>
- ☐ <javafx.controls>19.0.2.1</javafx.controls>

Maximum marks: 2

3 Controller - tómur klasi

Eftirfarandi **Controller** klasi er stýriklasi (controller) í JavaFX forriti sem hefur enga atburðahandlara (e. event handlers) eða atburðafiltera (e. event filters). Hvað er hægt að segja um notendaviðmótið?

```
package vidmot;  
public class Controller {  
    // kóði  
}
```

Veldu eitt af eftirtöldu

- ☐ Það er engin fx:controller settur í .fxml skrá
- ☐ Það er engin .fxml skrá sem lýsir notendaviðmótinu
- ☐ Klasinn bregst ekki við stýringu frá notanda
- ☐ Controller klasinn er eini klasinn í vidmot pakkanum (package)
- ☐ Forritið les ekki inn .fxml skrá

Maximum marks: 2

4 Vinnsluklasi

Hvert eftirfarandi er satt?

Veldu eitt af eftirtöldu

- ☐ Að hafa útreikninga hluta af notendaviðmóti einfaldar skipulag hugbúnaðarins
- ☐ Sérstakur viðmótsklasi, svokallaður stýriklasi (controller), inniheldur alla útreikninga og þannig fæst gott skipulag (arkitektúr)
- ☐ Útreikningar fara fram í viðmótsklösum (e. frontend) til að tryggja hraða og sveigjanleika
- ☐ Með því að hafa sérstakan vinnsluhluta (e. backend) sem sér um útreikninga er skipulag (arkitektúr) hugbúnaðarins gott og hver eining skýrt afmarkað hlutverk
- ☐ Sérstakur viðmótsklasi, svokallaður reikniklasi (compute class), stjórnar öllum útreikningum og þannig fæst klasi með skýrt afmarkað hlutverk

Maximum marks: 2

5 SliderProperty

Skoðið eftirfarandi forritsbút:

```
public Label fxLabel;  
public Slider fxSlider;  
public void initialize() {  
    fxSlider.setValue(24);  
    fxSlider.valueProperty().addListener((observable, oldvalue, newvalue) -> {  
        fxLabel.setText("Font value: " + newvalue.toString());  
        fxLabel.setStyle("-fx-font-size:" + newvalue.intValue() + ";" +  
            "-fx-background-color:black; "+  
            "-fx-text-fill:rgb(255,255,"+newvalue.intValue()/100.0*255+");");  
    });  
}
```

Hvaða viðmótshlutur breytist þegar eiginleiki **Slider** hlutarins fær nýtt gildi

Veldu eitt af eftirtöldu

- ☐ fxSlider
- ☐ fxLabel.textProperty()
- ☐ fxLabel
- ☐ Value
- ☐ Það verða engar breytingar því kóðinn er í initialize() aðferðinni

Maximum marks: 2

6 Sena - búa til

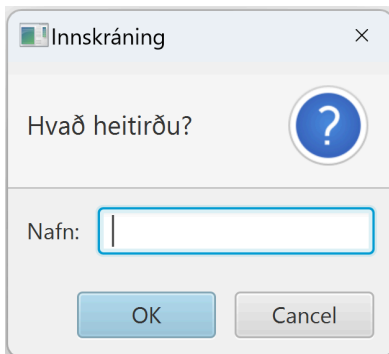
Sena (Scene) er búin til með því að:

Veldu eitt af eftirtöldu

- ☐ tengja controller við senu
- ☐ tengja hlut af Application klasa við senu
- ☐ birta (show) glugga með senu
- ☐ tengja rót viðmótstrés við senu
- ☐ tengja glugga við senu

Maximum marks: 2

7 TextInputDialog



Skoðaðu eftirfarandi kóða sem birtir TextInputDialog hér að ofan og safnar upplýsingum úr honum

```
@FXML
private Label fxNafn;
public void initialize() {
    TextInputDialog d = new TextInputDialog();
    d.setTitle("Innskráning");
    d.setHeaderText("Hvað heitirðu?");
    d.setContentText("Nafn:");
    Optional<String> utkoma = d.showAndWait();
    if(utkoma.isPresent()) {
        fxNafn.setText("nafnið er "+utkoma.get());
    }
    else {
        fxNafn.setText ("ekkert svar");
    }
    d.close();
}
```

Hvaða aðferð er kallað á sem gefur upplýsingar um hvort ýtt var á Cancel?

Veldu eitt af eftirtöldu

- ☐ isPresent()
- ☐ show()
- ☐ isCancel()
- ☐ isOk()
- ☐ setText (String s)

Maximum marks: 2

8 Sérhæfður klasi_2

Eftirfarandi skrá, **mynd-view.fxml**, lýsir noendaviðmóti fyrir sérhæfðan javaFX klasa, **Mynd**.

```
<?xml version="1.0" encoding="UTF-8"?>
<?import javafx.scene.image.*?>
<?import javafx.scene.layout.*?>
<fx:root onMouseExited="#fxMouseExited" type="javafx.scene.layout.AnchorPane"
prefHeight="400.0" prefWidth="184.0"
    xmlns="http://javafx.com/javafx/11.0.2"
xmlns:fx="http://javafx.com/fxml/1" >
    <ImageView fx:id="fxDyr" fitHeight="102.0" fitWidth="89.0" layoutX="14.0"
layoutY="173.0" pickOnBounds="true"
        preserveRatio="true" AnchorPane.bottomAnchor="14.0"
        AnchorPane.leftAnchor="14.0">
        <Image url="@myndir/kisa1.jpg"/>
    </ImageView>
</fx:root>
```

Ef þú ætlar að setja inn hlut af klasanum **Mynd** í aðra skrá **malverk-view.fxml** hvernig er það sýnt í skránni

Veldu eitt af eftirtöldu

- ☐ <Image fx:controller="Mynd" />
- ☐ <fx:root fx:id = "Mynd" />
- ☐ <fx:include "Mynd" />
- ☐ "Mynd"
- ☐ <Mynd />

Maximum marks: 2

9 menu include

Hvert eftirfarandi er satt um FXML búttinn hér á eftir

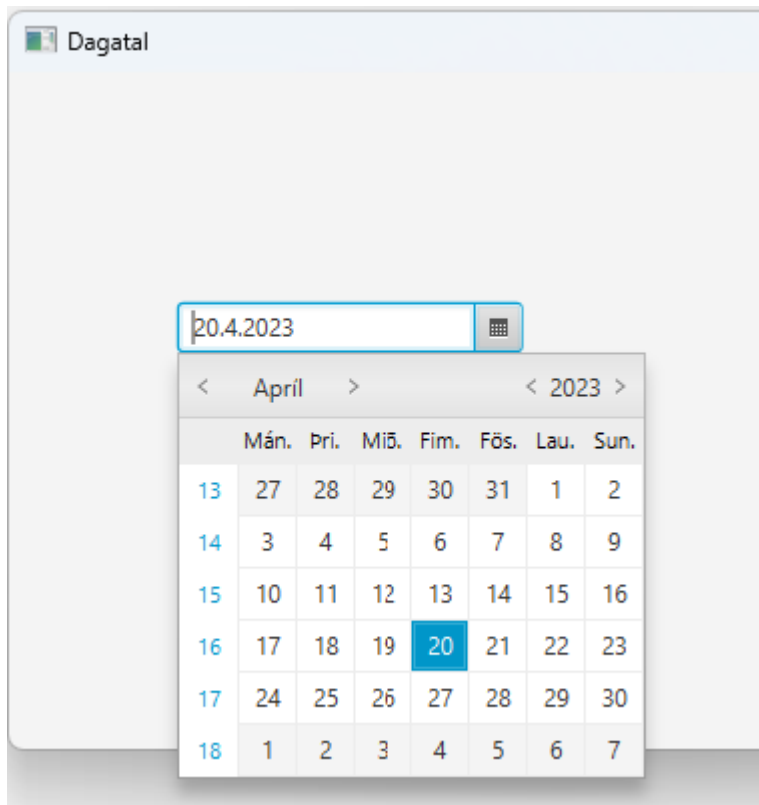
```
<VBox prefHeight="470.0" prefWidth="640.0"
xmlns="http://javafx.com/javafx/17.0.2-ea" xmlns:fx="http://javafx.com/fxml/1"
fx:controller="is.vidmot.GoldController">
  <children>
    <fx:include source="menu-view.fxml" fx:id="menuStyring" />
  </children>
</VBox>
```

Veldu eitt af eftirtöldu

- ☐ java þýðandi 17.0.2 er notaður til að þýða skrána
- ☐ Setningarnar skilgreina javaFX glugga með valmyndum (menu)
- ☐ Setningarnar skilgreina javaFX senu (scene) af æskilegri hæð 470 og æskilegri breidd 640
- ☐ Setningarnar skilgreina klasann GoldController
- ☐ Setningarnar setja inn (includes) FXML skrána menu-view.fxml með controller sem fer í breytuna menuStyringController í GoldController

Maximum marks: 2

10 DatePicker og Combobox



Hér að ofan er forrit í keyrslu með DatePicker klasanum.

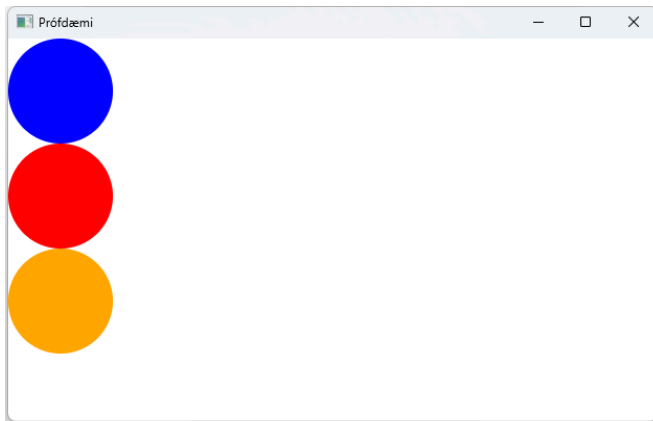
Hvað er sameiginlegt með **DatePicker** og **Combobox** klösunum?

Veldu eitt af eftirtöldu

- ☐ Þeir hafa LocalDate sem stök
- ☐ Þeir eru báðir container klasar og eru því ekki á laufum viðmótstrjáa
- ☐ Þeir eru báðir yfirklassar (superclass) ListView klasans
- ☐ Þeir erfa (extends) báðir frá ListView klasanum
- ☐ Þeir erfa (extends) báðir frá ComboboxBase<> klasanum

Maximum marks: 2

11 Hreyfing hringir



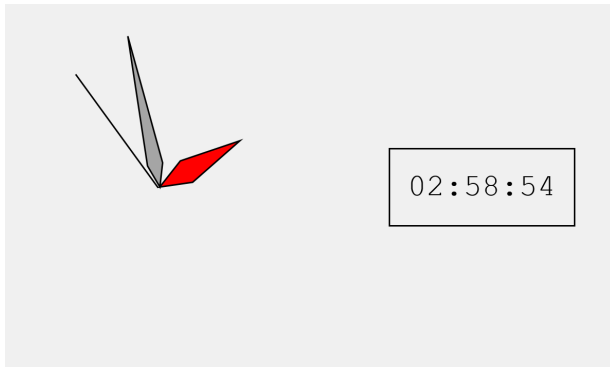
Skoðaðu eftirfarandi forrit sem byrjar í stöðunni hér að ofan. Hver eftirfarandi fullyrðinga er sönn eftir að forritið lýkur keyrslu?

```
@FXML
private Circle movingRED;
@FXML
private Circle movingOrange;
@FXML
private Circle movingBLUE;
public void initialize() {
    KeyValue[] keyValues = {
        new KeyValue(movingOrange.translateXProperty(), 300),
        new KeyValue(movingBLUE.translateXProperty(), 500),
        new KeyValue(movingRED.translateXProperty(), 100)
    };
    KeyFrame[] keyFrames = new KeyFrame[keyValues.length];
    for (int i = 0; i < keyValues.length; i++) {
        keyFrames[i] = new KeyFrame(Duration.seconds(50), keyValues[i]);
    }
    Timeline timeline = new Timeline(keyFrames[0], keyFrames[1],
keyFrames[2]);
    timeline.setCycleCount(1);
    // Spila tímalínuna
    timeline.play();
}
```

Veldu eitt af eftirtöldu

- ☐ Hringirnir hreyfast endalaust
- ☐ Blái hringurinn endar lengst til vinstri
- ☐ Rauði hringurinn hreyfist hraðast
- ☐ Rauði hringurinn endar lengst til vinstri
- ☐ Blái hringurinn hreyfist hægst

12 ClockControl Skin



Skoðaðu eftirfarandi kóðabút sem er hluti af forriti sem útfærir klukkuna hér að ofan

```
public ClockControl(@NamedArg("st") SkinType st) {  
    this.skinType = st;  
    Timeline ttimer = new Timeline(new KeyFrame(Duration.seconds(1),  
        (event) -> {  
            timeProp.setValue(new Date());  
        }));  
    ttimer.setCycleCount(Timeline.INDEFINITE);  
    ttimer.playFrom(Duration.millis(999));  
}
```

Hvaðan fær ClockControl upplýsingar um hvað klukkan er?

Veldu eitt af eftirtöldu

- ☐ Með því að smíða nýjan Date hlut þegar ClockControl hlutur er smíðaður
- ☐ Með því að smíða nýjan Date hlut á einnar sek fresti
- ☐ Með því að kalla á kerfisklukkuna sem fylgir innbyggða java forritasafninu
- ☐ Með því að frumstillta tímann sem þann tíma sem ClockControl hluturinn er smíðaður
- ☐ Með því að reikna út liðinn tíma síðann ClockControl hluturinn var frumstilltur

13 Notendakrafa

Hvert eftirfarandi er góð gagnvirkniskrafa

Veldu eitt af eftirtöldu

- ☐ Forritið verður gert hraðvirkara
- ☐ Forritinu er skipt í viðmóts og vinnslupakka
- ☐ Notandi getur stillt erfiðleikastig leiks, létt, miðlungs og erfitt
- ☐ Notandi sér 5 +/- 2 litir í notendaviðmótinu
- ☐ Notendaviðmótið verður endurbætt

Maximum marks: 2

14 Verkefni fyrir Tónlistarveitu

Hvert eftirfarandi er gott verkefni (e. activity) til að leggja fyrir notendur í notendaprófunum á forritinu Tónlistarveita

Veldu eitt af eftirtöldu

- ☐ Veldu lagið "Yfir strikið"
- ☐ Biddu vin þinn að spila lagið "Yfir strikið"
- ☐ Í snjallsíma veldur lagið "Yfir strikið" á Spotify
- ☐ Opnaðu nýjasta lagalistann þinn og veldu eitthvert lag
- ☐ Ýttu á lagið "Yfir strikið" í lagalista

Maximum marks: 2

15 A/B prófanir

Hvert er markmið A/B prófana?

Veldu eitt af eftirtöldu

- ☐ Til að safna gögnum um hegðun notenda og óskir til að nota til að besta hugbúnaðinn
- ☐ Notendum er skipt á slembinn hátt í tvo hópa og viðbrögð þeirra við mismunandi útgáfur af hugbúnaði eru borin saman
- ☐ Að fá tvo sérfræðinga í notendaviðmótum til að framkvæma sérfræðingspróf og bera saman svör þeirra
- ☐ Að finna marktækan mun tveggja notendahópa með spurningakönnun
- ☐ Að ákveða bestu útgáfu af hugbúnaði án þess að taka tillit til sjónarmiða notenda

Maximum marks: 2

i Inngangur að hluta 2

Hluti II

Í þessum hluta eru 6 spurningar sem gilda samtals 70%.

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.

16 Teiknimyndasaga

Leysið dæmið á skannblöðum.

Í þessu dæmi er sama sviðssetning og notuð er í öðru dæmi í prófinu. Dæmin eru þó óháð hvort öðru og hægt að leysa þau í hvaða röð sem er.

Skoðið eftirfarandi sviðssetningu (e. scenario) af bókunarhugbúnaði fyrir veitingastaði. Sjá resource #6 Hönnuður bókunarhugbúnaðarins vill fá betri skilning á hver gagnvirknin er og hvernig útlit notendaviðmótsins er.

Gerðu teiknamyndasögu með 4-6 römmum. Gefðu skýrt til kynna hvað af textanum í sviðssetningunni hér að ofan þú notar sem grunn fyrir teiknimyndasöguna. Þú gætir notað allan textann eða hluta af honum.

Undir hvern ramma skaltu setja texta sem lýsir honum og ef við á hvernig notandi fer í næsta ramma. Rammi á að vera teikning af notendaviðmóti eingöngu. Hvorki notandi né umhverfi hans á að sjást í rammanum. Teikning á að lýsa notendaviðmóti eins nákvæmlega og hægt er. Teiknimyndasaga á að hafa upphaf og endi og hafa gott flæði á milli ramma. Gefið ykkur tíma til að skipuleggja hvað á að vera í hverjum ramma.

Maximum marks: 10

17 Viðmótstré

Leysið dæmið á skannblaði.

Myndin hér að neðan er af JavaFX forriti sem hefur eftirfarandi gagnvirkni:

- 1) Þegar forritið er opnað er Ísland valið, íslenski fáninn er sýndur og íslenski þjóðsöngurinn spilaður
- 2) Notandi getur gert pásu á spilun (||) og haldið áfram að spila (>) Viðmótið sýnir || eða > eftir því hver staða spilunar er
- 3) Notandi getur valið land og þá er þjóðsöngur landsins leikinn frá byrjun

Teiknið **viðmótstré** fyrir eftirfarandi notendaviðmót. Sýnið klasa hlutanna og afstöðu þeirra í **viðmótstrénu**.

Merkið inn á viðmótstréð eftirfarandi upplýsingar, gjarnan með örvum inn í viðmótstré og skýringum í texta:

Sýnið hvaða hlutir (einn eða fleiri) þurfa að hafa **breytur** og gefið **breytunum nafn**

Sýnið hvaða hlutir (einn eða fleiri) þurfa að hafa **atburðahandlery** og gefið þeim nafn

Vísbending: Í næstsíðasta dæmi prófsins sést dæmi um viðmótstré



Maximum marks: 10

Í þessu dæmi er sama sviðssetning og notuð er í öðru dæmi í prófinu. Dæmin eru þó óháð hvort öðru og hægt að leysa þau í hvaða röð sem er.

Þú átt að skipuleggja notendaprófanir fyrir bókunarforrit fyrir veitingastaði **GoOut**. Hópurinn þinn heitir A og hefur 4 forritara/hönnuði. Hópurinn þinn hefur séð um hönnun og forritun á **GoOut**. Meðfylgjandi er stutt lýsing á sviðssetningu, sjá resource #6

Í fyrirtækinu eru tveir aðrir jafnstórir hópar, B og C, sem hafa líka forritara og hönnuði. Í fyrirtækinu er markaðsdeild, M, með tveimur starfsmönnum sem sjá um að skilja þarfir notenda, fá álit þeirra á vörunni og almennt um tengsl við notendahópinn.

Starfsmenn heita A1,..A4, B1,..B4, M1,.. o.s.frv.

Þar sem A hefur takmarkaða fjármuni viltu nota að helmingi til notendur innanhúss og að helmingi til notendur utan fyrirtækisins. Notendur utanhúss heita N1,.... eins marga og þú þarft á að halda.

a) Skipuleggðu hverjir taka þátt í notendaprófunum með því að fylla út í eftirfarandi töflu. Ef þú vilt geturðu bætt dálkum og röðum við töfluna. Tilgreindu greinilega hvaðan fólkíð kemur.

Rökstuddu fjölda umsjónarmanna og notenda sem taka þátt.

Rökstuddu val þitt á fólkinu í töflunni.

Nafn umsjónarmanns	Auðkenni notanda	Dagsetning	Tími	Staður

b) Skrifaðu 3-5 verkefni (e. activity) sem þú ætlar að nota og leggja fyrir notendur í notendaprófunum. Settu upp í svona töflu

Númer verkefnis	Verkefni og gögn
1	
2	
3	
4	
5	

¹⁸ Skipulagning notendaprófana

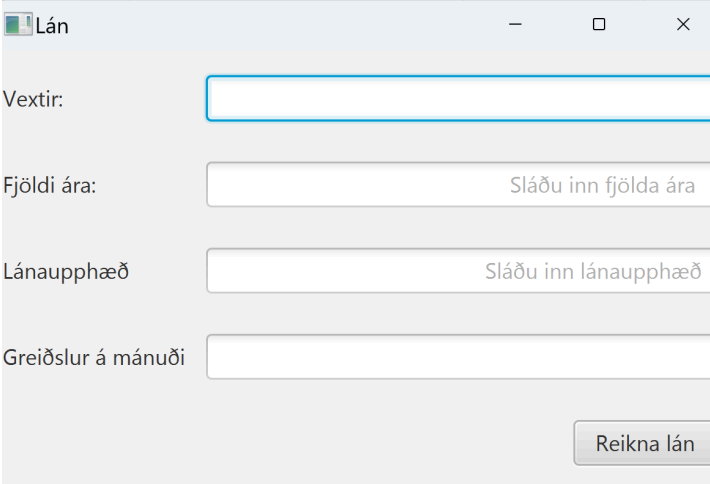
Skrifið svarið hér. Notið töflur. Merkið svörin við liði a) og b)

[illegible]

Maximum marks: 10

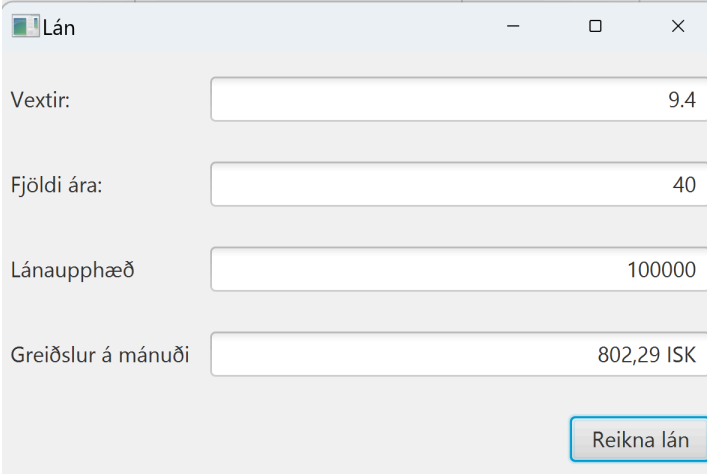
Þið eigið að forrita lánareiknivél. Notandi setur inn í textasviðin (TextField) vexti á láni, fjölda ára sem hann/hún/það ætlar að greiða lánið upp á og og upphæð láns. Notandi setur **ekki** inn <enter> í lokin á textasviðunum. Þegar notandi ýtir á hnappinn **Reikna lán** birtist í textasviðið greiðsla á mánuði. Ekki er hægt að breyta því textasviði.

Eftirfarandi mynd sýnir notendaviðmót eftir að forritið er keyrt upp



The screenshot shows a window titled "Lán" with four input fields and a button. The fields are labeled "Vextir:", "Fjöldi ára:", "Lánaupphæð", and "Greiðslur á mánuði". The "Fjöldi ára:" and "Lánaupphæð" fields have placeholder text "Sláðu inn fjölda ára" and "Sláðu inn lánaupphæð" respectively. The "Reikna lán" button is at the bottom right.

og eftirfarandi mynd sýnir notendaviðmót eftir að notandinn hefur sett inn gögnin 9.4, 40 og 100000 og ýtir á **Reikna lán**



The screenshot shows the same "Lán" window but with the input fields filled with values. "Vextir:" is 9.4, "Fjöldi ára:" is 40, "Lánaupphæð" is 100000, and "Greiðslur á mánuði" is 802,29 ISK. The "Reikna lán" button is now highlighted with a blue border.

Þið fáið vinnsluklasann **Lan** (resource #7) sem geymir vexti, fjölda ára og upphæð láns. Klasinn getur reiknað út greiðsluupphæð á mánuði. Þið eigið **ekki** að forrita klasann heldur nota hann.

Eftirfarandi er beinagrind af klasanum **LanController** sem er controller klasi fyrir .fxml skrá notendaviðmótsins. Ljúkið við að forrita klasann. Notið klasann **Lan** í LanController

```
package vidmot;
//import setningum er sleppt

public class LanController {

    // forritið tilviksbreytur
```

```
    // forritið aðferðir eftir þörfum  
}
```

Visbending:

aðferðin format á String er skilgreind svona
static String format (String format, Object... args)

og skilar sniðnum streng. Dæmi um notkun er
double tala = 1000.23;
String.format ("%8.1f", tala); // prentar út: 1000.2

19 Lan - Verkefni 2

Forritaðu hér

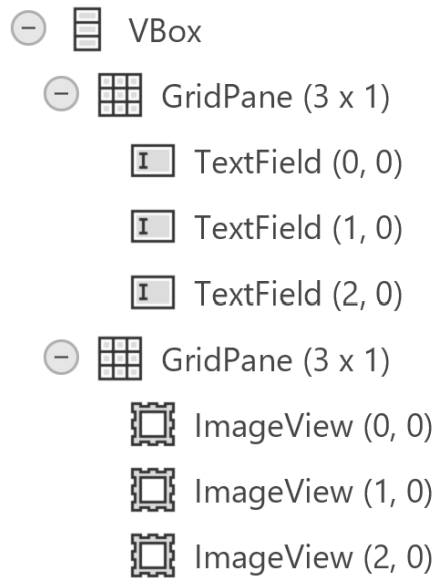
1	
---	--

Maximum marks: 10

Forritið í þessu dæmi leyfir notanda að víxla myndum með því að setja inn tölu

Vísbendingar eru neðst:

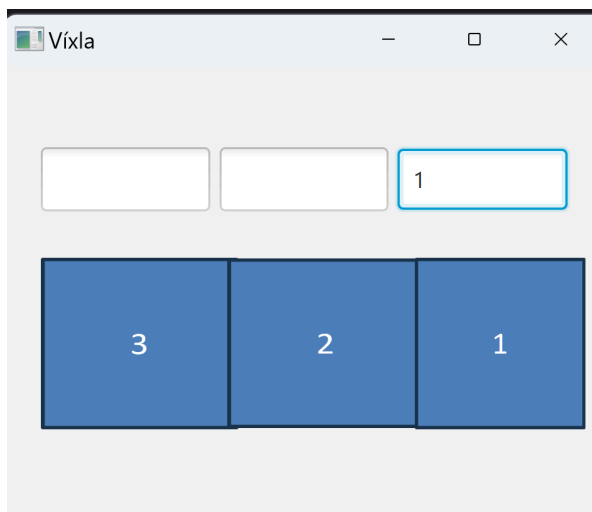
Viðmótstréð fyrir notendaviðmótið er hér á eftir



Eftirfarandi er upphafsmynd forritsins



Eftirfarandi mynd sýnir notendaviðmótið eftir að notandi hefur sett töluna 1 í reit 3. Búið er að víxla á myndum í dálkum 1 og 3.



Forritið tvær aðferðir, handlerinn **onTala** sem er handler fyrir TextField hlutina í efri GridPane

og hjálparaðferðina **vixla** eins og er sýnt í eftirfarandi beinagrind á **RodController.java**

Klárið að forrita **RodController.java** þannig þegar notandi setur inn tölu, **fra**, á bilinu 1-3 í reit númer **til** (TextField) og ýtir á <enter> þá er víxlað á myndum (Image) í reitum (ImageView) **fra** og **til**.

```
package vidmot;

// imports er sleppt - imports are omitted

public class RodController {

    @FXML
    protected GridPane fxRod; // neðri röð með þremur ImageView hlutum

    @FXML
    public void onTala(ActionEvent actionEvent) {
        TextField textField = (TextField)actionEvent.getSource();
        // forritið héðan - program here

        // kallið á vixla - call vixla

    }

    /**
     * Víxlar á myndum í reitum fra og til í neðri GridPane
     * @param fra heiltala sem er númer dálks í (neðri) GridPane með mynd
     * @param til heiltala sem er númer dálks í (neðri) GridPane með mynd
     */
    private void vixla(int fra, int til) {
        // forritið hér - program here
    }
}
```

Vísbending: Dálkar í GridPane er taldir frá 0.

Hægt er að ná í öll börn **Parent** hlutar, **p**, með **p.getChildren()**
Hægt er að finna í hvaða dálk hlutur **h** er í með static aðferðinni **getColumnIndex(h)**
GridPane.getColumnIndex(h)

Hægt er að ná í Image hlut úr ImageView hlut, **iv**, með aðferðinni **iv.getImage()**

20 Víxla - verkefni 3

Forritaðu hér

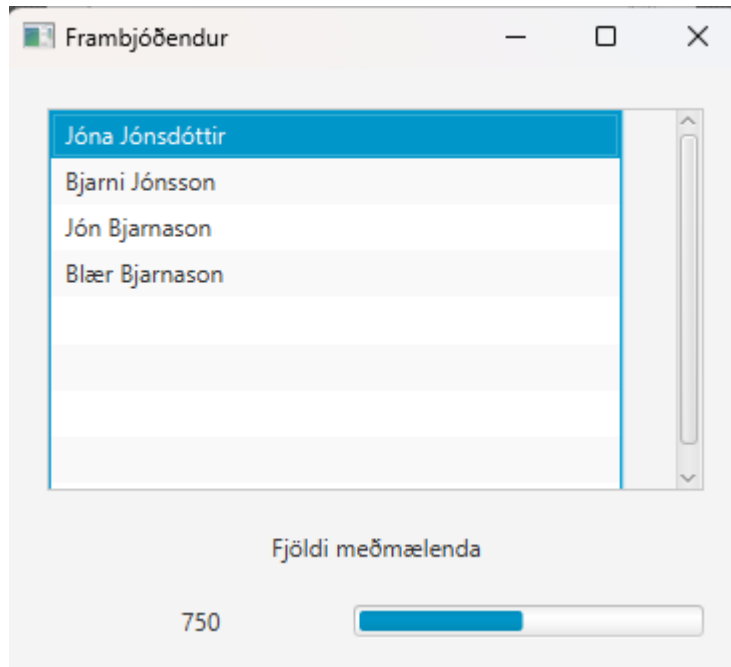
1

Maximum marks: 15

Vísbendingar eru neðst:

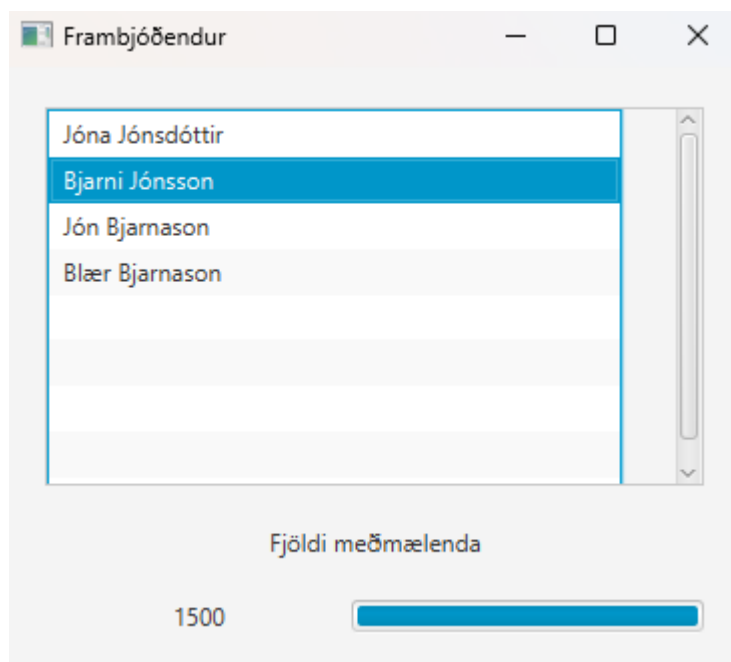
Forritið í þessu dæmi sýnir lista af frambjóðendum og fjölda meðmælanda sem þau hafa safnað.

Myndin hér á eftir sýnir upphafsmynd forritsins í keyrslu. Framvinduslái (e. progress bar) sýnir hlutfall framvindu þar sem 1500 er 100%.



Í upphafi er fyrsti frambjóðandinn valinn, fjöldi meðmælanda sem hann hefur safnað og framvinduslái samkvæmt því.

Þegar notandi ýtir með mús á eitthvert atriði í listanum uppfærast fjöldi meðmælanda fyrir neðan listann og framvinduslái einnig. Fjöldi meðmælanda er mest 1500.



Eftirfarandi skilgreining af klasa er gefin. **Ekki forrita klasann AllirFrambjodendur** en þið getið kallað á aðferðina lesaFrambod()

```
public class AllirFrambjodendur {  
  
    /**  
     * Skilar lista af Frambjodandi hlutum  
     * @return listi af Frambjodandi hlutum  
     */  
    public static ObservableList<Frambjodandi> lesaFrambod() {
```

Forritið **FrambjodandiController**. Eftirfarandi er beinagrind af klasanum.

```
package vidmot;  
// imports er sleppt - imports are omitted  
public class KosningController {  
  
    // skilgreinið tilviksbreytur    // define instance variables  
  
    // frumstilling á controller  
    public void initialize() {  
        // forritið hér - program here  
    }  
  
    // atburðahandler fyrir að velja atriði á lista  
    @FXML  
    private void onValinFrambjodandi(MouseEvent mouseEvent) {  
        // forritið hér - program here  
    }  
  
    // bætið við private hjálparaðferðum sem kallað er á úr öðrum  
    aðferðum  
    // add private utility methods that are used in other methods  
  
}
```

Forritið vinnsluklasann **Frambjodandi** sem hefur tilviksbreytuna **nafn** (strengur) og **fjoldiMedmaela** (int). Hér er beinagrind

```
package vinnsla;  
  
public class Frambjodandi {  
    // skilgreinið tilviksbreytur  
    // define instance variables  
  
    public Frambjodandi(String nafn, int fjoldiMedmaela) {  
        // forritið hér - program here  
    }  
  
    // Forritið aðrar aðferðir  
    // Program other methods  
}
```

Vísbendingar:

Viðmótsklasarnir sem eru notaðir til að lýsa notendaviðmótinu í .fxml skrá eru VBox, ScrollPane, ListView, Label, HBox, og ProgressBar

21 Frambjóðandi - verkefni 4

Forritaðu hér

1	
---	--

Maximum marks: 15