

.....

Хранится информация о домашнем животном: вид (кошка, собака, черепаха и т. п.), кличка, возраст – количество лет и месяцев, имя хозяина. В главном окне приложения отображается информация о животном в текстовом виде.

Реализуйте возможность увеличения возраста животного на один месяц при помощи кнопки в главном окне.

Для решения задачи используйте архитектуру Модель-Вид-Контроллер. Создайте отдельный класс для хранения данных о животном, в котором опишите метод увеличения возраста на один месяц с учетом того, что в году 12 месяцев и, например, при увеличении исходного возраста 2 года и 11 месяцев, текущее значение должно стать равно 3 года 0 месяцев. Для отображения данных создайте отдельный класс, в котором опишите несколько визуальных элементов для отображения данных. Свяжите этот класс-вид с объектом-моделью. В главном классе приложения создайте соответствующие объекты и организуйте взаимодействие с пользователем для увеличения возраста животного.

В поле ответа на задание введите название элемента архитектуры приложения: модель, вид или контроллер (без пробелов!), к которому будет относиться класс, хранящий информацию о животном.

Дополнительно: добавьте к информации о животном его изображение и выводите его вместе с остальными данными; осуществляйте чтение исходных данных из текстового файла.

.....

Отсылка на 13 ЛБ

Изменим функциональность приложения, работающего с премиальным фондом организации. Добавим пользователю возможность изменять название организации и величину премии.

Приведём полный код класса модели:

```
import java.time.LocalDate ;

import javafx.beans.property.DoubleProperty ;

import javafx.beans.property.SimpleDoubleProperty ;

import javafx.beans.property.SimpleStringProperty ;

import javafx.beans.property.StringProperty ;

public class Organization {

    private StringProperty name ;
```

```

public StringProperty nameStringProperty () {

    if (name == null) {

        name = new SimpleStringProperty();

    }

    return name; }

public final void setName (String value) {

    nameStringProperty().set(value); }

public final String getName () {

    return nameStringProperty().get(); }

private int personnel;

private String holiday;

private LocalDate date;

private DoubleProperty bonus;

public DoubleProperty bonusProperty () {

    if (bonus == null) {

        bonus = new SimpleDoubleProperty();

    }

    return bonus; }

public final void setBonus (Double value) {

    bonusProperty().set(value); }

public final Double getBonus () {

    return bonusProperty().get(); }

public Organization (String name, int personnel, String holiday, LocalDate
date, double bonus) {

    setName (name);

    this.personnel = personnel;

    this.holiday = holiday;

    this.date = date;

```

```
        setBonus (bonus) ; }

    public int getPersonnel () {

        return personnel ; }

    public String getHoliday () {

        return holiday ; }

    public LocalDate getDate () {

        return date ; } } }
```

Код класса вид:

```
import java.time.format.DateTimeFormatter ;

import javafx.beans.binding.Bindings ;

import javafx.geometry.HPos ;

import javafx.geometry.Insets ;

import javafx.geometry.Pos ;

import javafx.scene.control.Label ;

import javafx.scene.layout.GridPane ;

import javafx.scene.text.Font ;

import javafx.scene.text.FontWeight ;

import javafx.scene.text.Text ;

public class ViewOrganization {

    private Organization org ;

    private GridPane grid ;

    private Text nameOrg ;

    private Text holidayOrg ;

    private Text cashBonus ;

    private void createPane () {
```

```

grid = new GridPane();

grid.setAlignment(Pos.CENTER);

grid.setHgap(10);

grid.setVgap(10);

grid.setPadding(new Insets(25, 25, 25, 25));

nameOrg = new Text();

nameOrg.setFont(Font.font("Tahoma", FontWeight.NORMAL, 20));

GridPane.setHalignment(nameOrg, HPos.CENTER);

grid.add(nameOrg, 0, 0, 2, 1);

holidayOrg = new Text();

holidayOrg.setFont(Font.font("Tahoma", FontWeight.NORMAL, 20));

grid.add(holidayOrg, 0, 1, 2, 1);

Label cashBonusTitle = new Label("Bonus");

cashBonusTitle.setFont(Font.font("Tahoma", FontWeight.NORMAL,
20));

grid.add(cashBonusTitle, 0, 2);

cashBonus = new Text();

cashBonus.setFont(Font.font("Tahoma", FontWeight.NORMAL, 20));

grid.add(cashBonus, 1, 2); }

public GridPane getPane() {

    return grid; }

public void setOrganization (Organization org) {

    this.org = org;

    nameOrg.textProperty().bind(this.org.nameStringProperty());

```

```

        holidayOrg.setText(org.getHoliday()+" - "
+org.getDate().format(DateTimeFormatter.ofPattern("dd.MM.uuuu")));

        cashBonus.textProperty().bind(this.org.bonusProperty().asString())

        public ViewOrganization (Organization org) {

            createPane();

            setOrganization(org); }}

```

Контроллером остаётся главный класс приложения.

Приведём полный код главного класса приложения:

```

import java.time.LocalDate;

import javafx.application.Application;

import javafx.event.ActionEvent;

import javafx.geometry.Insets;

import javafx.geometry.Orientation;

import javafx.scene.Scene;

import javafx.scene.control.Button;

import javafx.scene.control.Label;

import javafx.scene.control.Separator;

import javafx.scene.control.Spinner;

import javafx.scene.control.TextField;

import javafx.scene.layout.HBox;

import javafx.scene.layout.VBox;

import javafx.scene.text.Font;

import javafx.stage.Stage;

public class Main extends Application {

    @Override

    public void start (Stage primaryStage) {

```

```

primaryStage.setTitle("Simple Model-View Data Binding");

Organization org = new Organization("Horns&Hoof", 10, "International
Women's Day", LocalDate.of(2016, 3, 9), 0);

HBox root = new HBox(10);

ViewOrganization viewOrg = new ViewOrganization(org);

root.getChildren().add(viewOrg.getPane());

root.getChildren().add(new Separator(Orientation.VERTICAL));

root.getChildren().add(editBlock(org));

Scene scene = new Scene(root, 900, 300);

primaryStage.setScene(scene);

primaryStage.show(); }

private VBox editBlock (Organization org) {

    VBox editPane = new VBox(10);

    editPane.setPadding(new Insets(20));

    Label labelTitle = new Label("Enter name organization\n and bonus");

    labelTitle.setFont(Font.font(20));

    TextField editName = new TextField();

    editName.setFont(Font.font(20));

    editName.setPrefSize(150, 40);

    Spinner<Double> editBonus = new Spinner<>(0, 100, 0, 0.5);

    editBonus.setPrefSize(100, 40);

    editBonus.setStyle("-fx-font-size: 20");

    Button btn = new Button("Edit");

    btn.setFont(Font.font(20));

    btn.setOnAction((ActionEvent event) -> {

```

```
        org.setName(editName.getText());

        org.setBonus(editBonus.getValue());    });

    editPane.getChildren().addAll(labelTitle, editName, editBonus,
btn);

    return editPane;    }

public static void main(String[] args) {

    launch(args);    }}
```