**Московский государственный технический**

**университет им. Н.Э. Баумана**

Факультет «Радиотехнический»

Кафедра «Информатика и вычислительная техника»

Курс «Парадигмы и конструкции языков программирования»

Отчет по лабораторной работе №6

|  |  |  |
| --- | --- | --- |
| Выполнил: |  | Проверил: |
| студент группы РТ5-31: |  | преподаватель каф. ИУ5 |
| Слкуни Г.Г. |  | Гапанюк Ю.Е.. |
| Подпись и дата: |  | Подпись и дата: |

Москва, 2023

**Текст программы:**

Класс Main:

package me.author.bot;  
  
import me.author.bot.manager.BotManager;  
import me.author.bot.manager.LogManager;  
import me.author.bot.model.Bot;  
  
import java.util.ArrayList;  
import java.util.List;  
  
public class Main {  
  
 private static final String *token* = "";  
 private static final long *ownerId* = 1;  
 public static List<String> *buttons* = new ArrayList<>();  
  
 public static void main(String[] args) {  
  
 *buttons*.add("Проверочная кнопка");  
 *buttons*.add("Текущая дата");  
 LogManager.*init*();  
  
 LogManager.*log*("Инициализируем бота...");  
 Bot bot = BotManager.*registerBot*(*token*);  
  
 LogManager.*log*("Отправляем тестовое сообщение...");  
 bot.sendMessage(*ownerId*, "Я запустился", *buttons*);  
  
 }  
  
}

Класс Bot:

package me.author.bot.model;  
  
import com.pengrad.telegrambot.TelegramBot;  
import com.pengrad.telegrambot.UpdatesListener;  
import com.pengrad.telegrambot.model.Message;  
import com.pengrad.telegrambot.model.Update;  
import com.pengrad.telegrambot.model.request.ReplyKeyboardMarkup;  
import com.pengrad.telegrambot.request.SendMessage;  
import me.author.bot.Main;  
  
import java.text.SimpleDateFormat;  
import java.util.ArrayList;  
import java.util.Date;  
import java.util.List;  
  
public class Bot {  
   
 private TelegramBot telegramBot;  
 private String token;  
  
 public Bot(String token) {  
 telegramBot = new TelegramBot(token);  
 this.token = token;  
  
 telegramBot.setUpdatesListener(new BotListener());  
  
 }  
  
 public TelegramBot getTelegramBot() {  
 return telegramBot;  
 }  
  
 public String getToken() {  
 return token;  
 }  
  
 public void sendMessage(long userId, String message) {  
 sendMessage(userId, message, new ArrayList<String>());  
 }  
  
 public void sendMessage(long userId, String message, List<String> buttons) {  
 SendMessage request = new SendMessage(  
 userId,  
 message  
 );  
 ReplyKeyboardMarkup keyboard = null;  
 if (buttons.size() > 0) {  
 keyboard = new ReplyKeyboardMarkup(buttons.get(0));  
 for (int i = 1; i < buttons.size(); i++) {  
 keyboard.addRow(buttons.get(i));  
 }  
 }  
 request.replyMarkup(keyboard);  
 telegramBot.execute(request);  
 }  
  
 private class BotListener implements UpdatesListener {  
  
 @Override  
 public int process(List<Update> list) {  
  
 for (Update update : list) {  
  
 Message message = update.message();  
 if (message != null) {  
 String text = message.text();  
 switch (text.toLowerCase()) {  
 case "проверочная кнопка": {  
 sendMessage(message.chat().id(), "Все работает!", Main.*buttons*);  
 return *CONFIRMED\_UPDATES\_ALL*;  
 }  
 case "текущая дата": {  
 sendMessage(message.chat().id(), String.*format*("Текущая дата: %s", new SimpleDateFormat("dd/MM/yyyy").format(new Date())), Main.*buttons*);  
 return *CONFIRMED\_UPDATES\_ALL*;  
 }  
 }  
 sendMessage(message.chat().id(), "Неизвестная операция :с", Main.*buttons*);  
 }  
  
 }  
  
 return *CONFIRMED\_UPDATES\_ALL*;  
  
 }  
 }  
  
}

Класс BotManager:

package me.author.bot.manager;  
  
import me.author.bot.model.Bot;  
  
import java.util.ArrayList;  
import java.util.List;  
  
public class BotManager {  
  
 private static final List<Bot> *bots* = new ArrayList<>();  
  
 public static Bot registerBot(String token) {  
 LogManager.*log*("Создаем нового бота...");  
 Bot bot = new Bot(token);  
 *bots*.add(bot);  
 return bot;  
 }  
  
}

Интерфейс ILog:

package me.author.bot.model.log;  
  
public interface ILog {  
  
 void Log(String prefix, String text);  
  
}

Класс ConsoleLog:

package me.author.bot.model.log;  
  
public class ConsoleLog implements ILog {  
  
 @Override  
 public void Log(String prefix, String text) {  
 System.*out*.printf("[%s]: %s%n", prefix, text);  
 }  
}

Класс LogManager:

package me.author.bot.manager;  
  
import me.author.bot.model.log.ConsoleLog;  
import me.author.bot.model.log.ILog;  
  
import java.util.ArrayList;  
import java.util.List;  
  
public class LogManager {  
  
 private static final List<ILog> logs = new ArrayList<>();  
  
 public static void init() {  
 logs.add(new ConsoleLog());  
 }  
  
 public static void log(String prefix, String message) {  
 for (ILog log : logs)  
 log.Log(prefix, message);  
 }  
  
 public static void log(String message) {  
 for (ILog log : logs)  
 log.Log("INFO", message);  
 }  
  
}

**Результат выполнения:**



