Лабораторная работа N1 "Базовые средства разработки для языка Java"

1. Цель работы – формирование комфортного окружения для разработки ПО на языке Java.

2. Исходные данные

Для выполнения лабораторных работ потребуется jdk последней версии, установленный на ваш компьютер и на удаленный сервер, а так же среда разработки IntelliJ IDEA Community Edition.

3. Задание – Лабораторная работа состоит из нескольких частей

- 1) Установка jdk на ваш компьютер
- 2) Установка jdk на удаленный сервер
- 3) Проверка работоспособности jdk
- 4) Установка IntelliJ IDEA на ваш компьютер

3.1

Установим jdk на ваш компьютер. Для этого переходим по этой ссылке на офиц. сайт

https://www.oracle.com/technetwork/java/javase/downloads/jdk13-downloads-5672538.html

Затем принимаем Accept License Agreement и выбираем третий

You must accept the Oracle Technology Network License Agreement for Oracle Java SE to download this software. Accept License Agreement Product / File Description File Size Download Linux 155.72 MB	Java SE Development Kit 13.0.2			
download this software. Accept License Agreement Product / File Description File Size Download Linux 155.72 MB jdk-13.0.2_linux-x64_bin.deb	You must accept the Oracle Technology Network License Agreement for Oracle Java SE to			
Operation Accept License Agreement ● Decline License Agreement Product / File Description File Size Download Linux 155.72 MB ±jdk-13.0.2_linux-x64_bin.deb	download this software.			
Product / File DescriptionFile SizeDownloadLinux155.72 MB€jdk-13.0.2_linux-x64_bin.deb	Accept License Agreement Internet Agreement			
Linux 155.72 MB 🛓 jdk-13.0.2_linux-x64_bin.deb	Product / File Description	File Size	Download	
	Linux	155.72 MB	jdk-13.0.2_linux-x64_bin.deb	
Linux 162.66 MB 🛓jdk-13.0.2_linux-x64_bin.rpm	Linux	162.66 MB	jdk-13.0.2_linux-x64_bin.rpm	
Linux 179.41 MB 🚽 jdk-13.0.2_linux-x64_bin.tar.gz	Linux	179.41 MB	jdk-13.0.2_linux-x64_bin.tar.gz	
macOS 173.3 MB ±jdk-13.0.2_osx-x64_bin.dmg	macOS	173.3 MB	jdk-13.0.2_osx-x64_bin.dmg	
macOS 173.7 MB 🚽 jdk-13.0.2_osx-x64_bin.tar.gz	macOS	173.7 MB	jdk-13.0.2_osx-x64_bin.tar.gz	
Windows 159.83 MB ±jdk-13.0.2_windows-x64_bin.exe	Windows	159.83 MB	jdk-13.0.2_windows-x64_bin.exe	
Windows 178.99 MB ±jdk-13.0.2_windows-x64_bin.zip	Windows	178.99 MB	jdk-13.0.2_windows-x64_bin.zip	

по счету файл (179.41 MB) Далее к нам скачается Архив , распаковываем его в нашу папку "~/", так , чтобы в нашей директории лежала директория jdk-13.0.2 .

3.2

Установим jdk на удаленный сервер. Я буду показывать на примере своей конфигурации, созданной для SSH соединения с сервером. Мое имя для конфигурации testjava, значит везде, где вы будете видеть testjava, изменяйте его на свое имя.

magamedrasul@pop-os:~\$ scp -r jdk-13.0.2 testjava@lab2.posevin.com:/home/testjava
The authenticity of host 'lab2.posevin.com (185.20.226.174)' can't be established.
ECDSA key fingerprint is SHA256:s6dIhfuNkpj1kYr7XkI0kMzJ+YW8P8iimve7950IXGY.
Are you sure you want to continue connecting (yes/no)? yes

Введите команду в своей папке:

scp -r jdk-13.0.2 testjava@lab2.posevin.com:/home/testjava

Далее вы должны подтвердить ключ для передачи файлов и соединения. Введите пароль (он не отображается в терминале). Все директория должна быть перемещена на ваш удаленный сервер.

У себя на компьютере в главной директории находим файл .bashrc . Заходим в него и пишем в конце

export JAVA_HOME=\$HOME/jdk-13.0.2 export PATH=\$JAVA_HOME/bin:\$PATH

Так же добавляем эти же команды в файл .profile (нужно перезагрузить компьютер после сохранения)

Вводим в терминале javac -version и нам должна выводиться текущая версия jdk.

Далее создаем файл Factorial.java, который содержит в себе следующий код:



Сохраняем файл, и в этой же директории в терминале пишем javac Factorial.java

Это позволит откомпилировать наш файл, затем мы получим файл с расширением .class . Мы должны запустить его с помощью :

java Factorial 5

Нам должна выводиться строка – 120 . Это результат выполнения программы

Доп. Шаг – скинем этот файл на сервер , так же как показывал до этого .

scp Factorial.java testjava@lab2.posevin.com:/home/testjava

3.3

Теперь нам надо войти в свою конфигурацию (так же не забывайте менять свое имя на testjava в моем примере)

ssh testjava@lab2.posevin.com

Вводим пароль. Теперь нам нужно настроить jdk на сервере. Для этого в терминале пишем export JAVA_HOME=/home/testjava/jdk-13.0.2 export PATH=\$JAVA_HOME/bin:\$PATH (не стоит забывать, что настраивать среду Java на удаленном

сервере придется каждый раз при подключении, так как на сервере менять .bashrc нельзя)

Теперь так же проверяем версию jdk и пробуем откомпилировать и запустить нашу программу Factorial на сервере.

```
testjava@lab2:~$ export JAVA_HOME=/home/testjava/jdk-13.0.2
testjava@lab2:~$ export PATH=$JAVA_HOME/bin:$PATH
testjava@lab2:~$ javac -version
javac 13.0.2
```

testjava@lab2:~\$ javac Factorial.java testjava@lab2:~\$ java Factorial 5 120

3.4

Установим среду разработки IntelliJ IDEA на компьютер

Скачиваем дистрибутив с сайта https://www.jetbrains.com/ruru/idea/download/#section=linux (берем бесплатную Community Edition)

Распаковываем в домашнюю папку, заходим в нее, находим папку bin. В ней есть файл idea.sh, запускаем его. Так как IDEA запускается первый раз, появится диалоговое окно, в котором нужно выбрать пункт Create New Project, после чего откроется окно создания нового проекта. Создание проекта – необходимое условие работы с интегрированной средой разработки, потому что проект, объединяющий набор файлов с исходным текстом программы, является в ней единицей сборки и выполнения.

В окне создания нового проекта, прежде всего, укажите путь к каталогу, в который вы установили JDK. Для этого нажмите кнопку New, в появившемся выпадающем списке нажмите на пункт JDK и в окне выбора каталога перейдите на каталог jdk-13.0.2, расположенный в вашем домашнем каталоге. IDEA будет использовать именно указанную вами версию JDK, даже если на вашем компьютере присутствуют другие версии. Два раза нажмите на кнопку Next и введите название вашего проекта – FactorialProject.

Отметим, что IDEA создаст для вашего проекта каталог ~/IdeaProjects/FactorialProject. Именно там нужно будет, в случае чего, искать ваши файлы.

Для создания проекта нажмите кнопку Finish, в результате чего откроется главное окно среды разработки. В левой части окна располагается дерево наших проектов, содержащее проект FactorialProject. Так как мы собираемся использовать возможности Java 13.0.2, а именно – замыкания, мы должны включить их поддержку для нашего проекта. Для этого выберите пункт меню File|Project Structure, выберите «13 – No new language features» в выпадающем списке Project Language Level и нажмите OK.

Настало время для добавления файлов в наш проект. Для этого раскройте его в дереве проектов, перейдите на пункт src и в контекстном меню выберите New|Java Class. Появится диалоговое окно с предложением ввести имя класса. Введите

Factorial и нажмите OK. После этого IDEA создаст файл с именем Factorial.java и откроет его в окне редактора. Замените в редакторе «заготовку» класса Factorial, которую создала для нас IDEA, на код из пункта

Для сборки нашего проекта выберите пункт меню Build|Make Project. Если проект не содержит ошибок, в левом нижнем углу главного окна IDEA появится надпись «Compilation completed successfully in x seconds». В противном случае откроется окно сообщений компилятора с описанием ошибок.

Параметры запуска нашего проекта в IDEA определяются так называемыми конфигурациями. Диалговое окно для работы с конфиграциями открывается путём выбор пункта меню

Run|Edit Configurations. Для создания новой конфигурации нажмите на кнопку с изображением знака «+» и в выпадающем списке выберите Application. После этого IDEA создаст новую безымянную конфигурацию и даст нам возможность редактировать параметры этой конфигурации. Дайте новой конфигурации имя FactorialApp, в качестве главного класса (Main class) укажите Factorial и задайте число, факториал которого хотите посчитать, в аргументах командной строки (Program arguments). Затем нажмите OK.

Теперь можно запустить проект, выбрав в меню Run|FactorialApp. В нижней части главного окна IDEA покажет содержимое стандартного потока вывода нашей программы.

Чтобы добавить программу на рабочий стол нажмите Tools|Create Desktop Entry