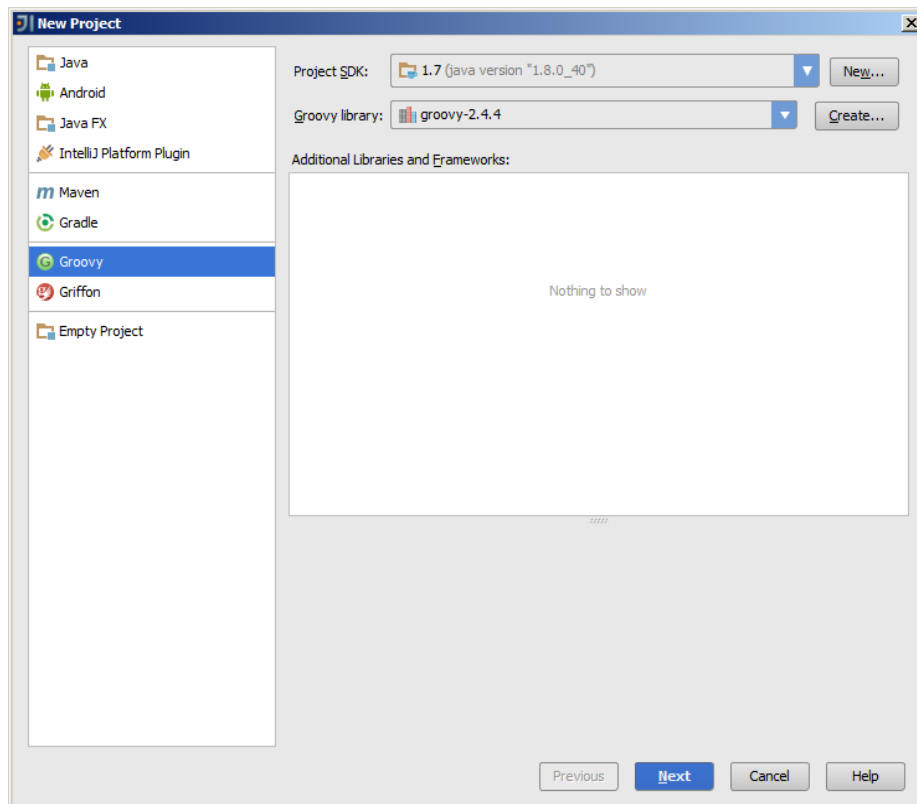


Урок 1

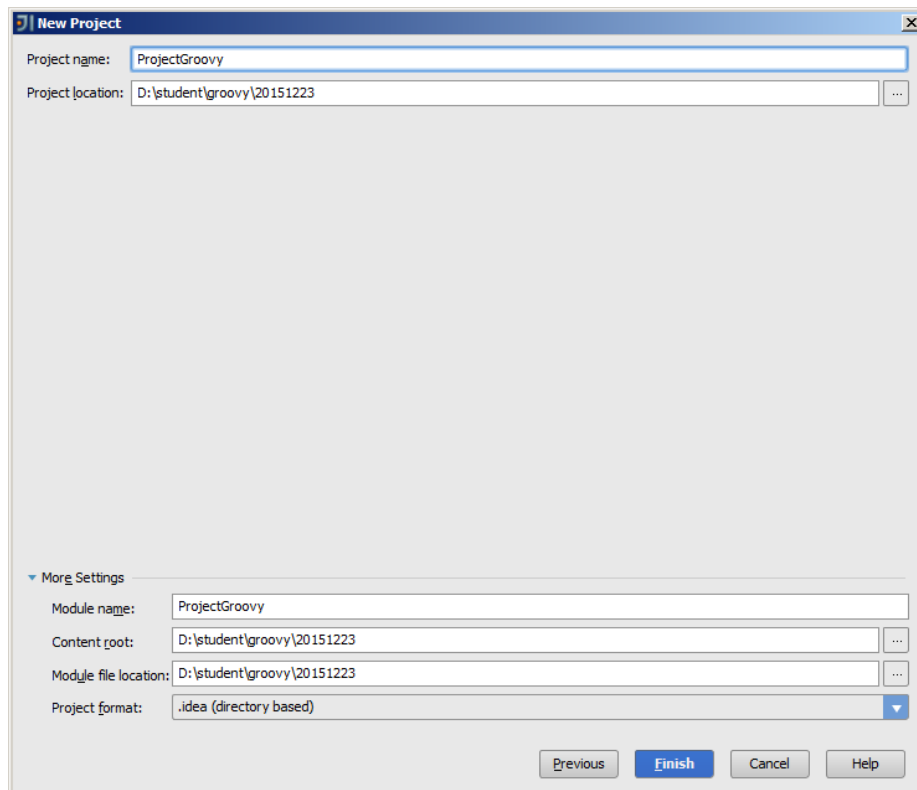
1. Установить Groovy на локальной машине
 - a. Проверить установку JDK или JRE версии не ниже 1.7
 - b. Запустить файл `groovy-2.4.4-installer.exe` и выбрать директорию для установки Groovy (далее – ДиректорияG).
 - c. Создать переменную окружения `GROOVY_HOME` = ДиректорияG
 - d. Добавить `%GROOVY_HOME%\bin` в переменную окружения `PATH`
 - e. Для проверки правильности установки, выполнить команду `groovy -v`
2. В Idea создать проект с именем `ProjectGroovy` для работы с Groovy



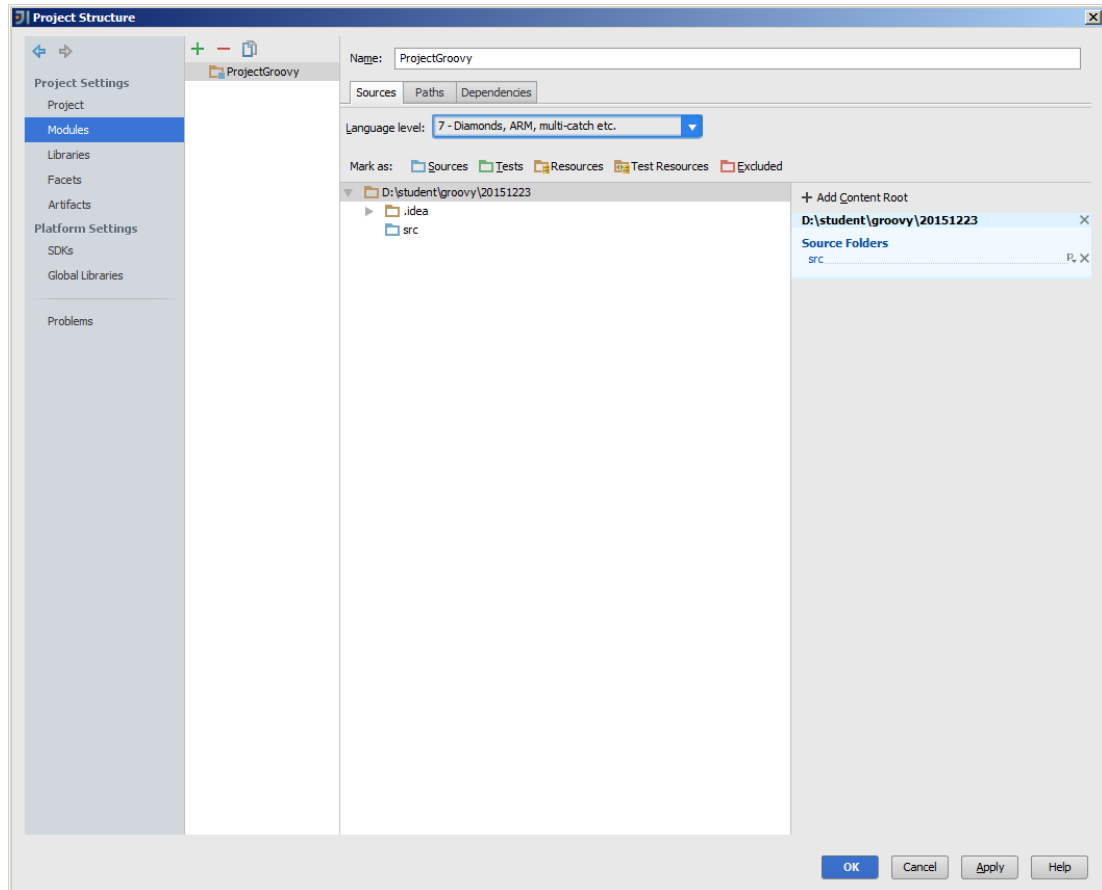
Выбрать строку с Groovy:



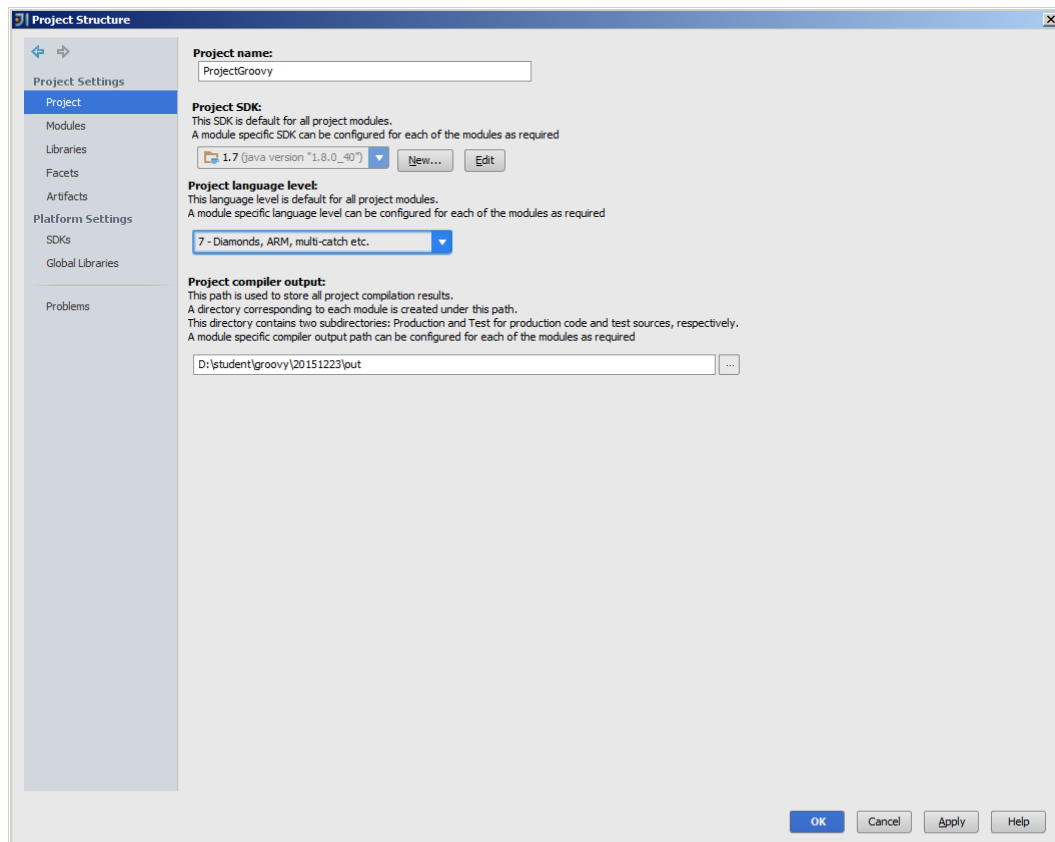
Указать имя проекта и его местоположение:



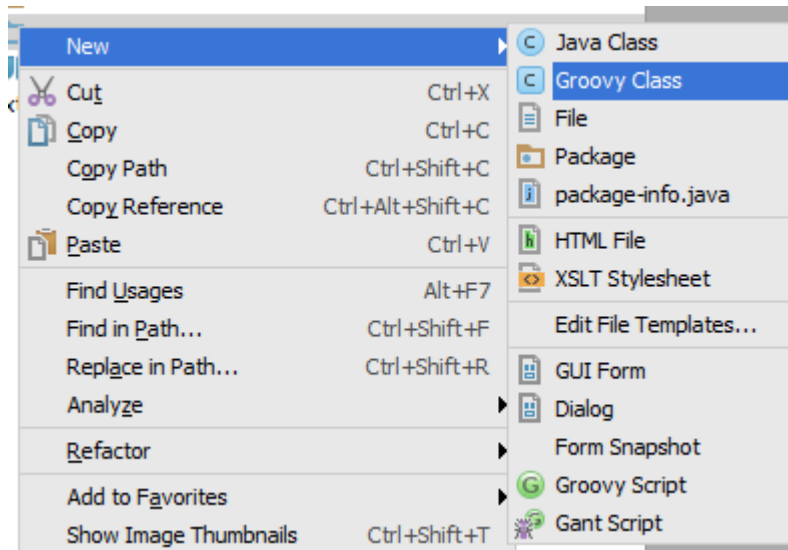
В настройках проекта выбрать уровень языка = 7



Также проверить этот уровень на вкладке Project -



3. Создать Groovy-класс (имя класса выберите самостоятельно):



4. В этом классе создайте метод main (как в Java) с таким кодом:

```
println "Welcome, Groovy"
```
5. Выполните этот метод и добейтесь его безошибочной работы.
6. Удалите из класса весь код (в том числе и заголовок класса) , оставив только строку

```
println "Welcome, Groovy"
```

7. Изменилось ли что-нибудь в результатах работы?
8. Проверка работы с конструкторами.
 - a. В созданном ранее классе `FirstClass` создать несколько переменных уровня экземпляра класса (имена и типы выберите сами). Конструкторов в этом классе создавать пока не требуется.
 - b. Создайте класс `Groovy` с именем `MainClass` и удалите из файла этого класса весь текст.
 - c. В файле `MainClass.groovy` создайте экземпляр класса `FirstClass` и попытайтесь инициализировать его конструктором с указанием параметров по имени, передав им произвольные значения
 - d. В классе `FirstClass` создайте явный конструктор и попытайтесь в файле `MainClass` создать экземпляр класса `FirstClass`. Возможно ли это сделать с поименной передачей параметров в этот конструктор.
9. При помощи экземпляра класса `Binding` задайте значение переменных экземпляра класса `FirstClass` и получите это значение. Будут ли видны эти значения при помощи `get`- и `set`-методов класса `FirstClass` ?
10. Проверьте при помощи оператора `instanceof` отношения между экземпляром класса `String` и классом `Integer`. Сделайте то же самое в Java-классе. В чем разница в операторе `instanceof` для `Groovy` и `Java` ?
11. Проверить необходимость оператора `def` и явного указания типа для «сокрытия» переменных уровня экземпляра класса от метода. Сделайте это в «явном» классе и «неявном».
12. Создайте метод с возвращаемым значением типа `Integer`, но без оператора `return`. Что надо сделать, чтобы метод выполнялся без ошибок?

Урок 2

1. Создайте два метода, один из которых возвращает тип `Integer`, а второй – тип `int`. В обоих методах должен быть один параметр типа `Integer`. Попытайтесь выполнить эти методы, передав в качестве параметра значение `null`. Какой из методов выполнится?
2. Создайте два экземпляра класса `BigDecimal` при помощи оператора `new`. Присвойте этим переменным одинаковые значения (с использованием конструктора). Убедитесь, что эти переменные не указывают на одно и то же место в памяти. Сравните значения переменных при помощи метода `equals`. Какие методы можно использовать для сравнения?
3. С этими же переменными попробуйте использовать методы `add`, `divide` и `mult`. Сравните результат этих операций с использованием обычных операндов `+`, `/` и `*`. Можно ли сделать то же самое в `Java`?

4. Создайте переменную без указания типа и присвойте ей значение 10. Получится ли присвоить этой же переменной значение 'f' или значение 'First'. Проверьте тип этой переменной при помощи метода `getClass().getName()`. Попробуйте поменять последовательность этих присвоений – сперва 'f', потом 'First', потом 10 и т.д.
5. Создайте две переменные типа `Date` и присвойте им значения 28 февраля 2015 года и 31 января 2015 года. Вычтите одну переменную из другой, а потом вычтите из 28 февраля 1 месяц, а затем прибавьте к результату 1 день и 1 месяц (это можно сделать двумя операциями). Что получилось?

Урок 3

1. Создать `Closure` для деления двух чисел. (a/b)
2. Создать `Closure` для вычитания двух чисел (d-c)
3. Создать `Closure` для выполнения операции a/b –c, используя две предыдущих `Closure`.

Урок 4

1. Подключиться из-под `Groovy` к базе данных `Oracle` (данные подключения высылались) под пользователем `hrXX` (01-12).
2. Выполнить запрос к таблице `employees` и сохранить результат в коллекции.
3. Проверить количество строк в коллекции.