**LAPORAN PRAKTIKUM OOP DASAR**

**MODUL 2**

**METHODS, VARIABLES, AND PARAMETERS**



**Adam Arthur Faizal**

**M3119001**

**TI A**

**PROGRAM STUDI TEKNIK INFORMATIKA**

**FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM**

**UNIVERSITAS SEBELAS MARET**

**SURAKARTA**

**2020**

**MODUL 2. Methods, Variables, and Parameters**

**Capaian Pembelajaran Praktikum:**

* Mengeksplorasi method dari library
* Mendefinisikan parameter & penggunaannya.
* Mengimplementasikan inheritance
* Mendeskripsikan property objek (variabel)
* Mendefinisikan & menerapkan konsep & terminology terkait.

**Tools:**

* Java Development Kit (JDK)
* Greenfoot IDE

**Terminologi:**

Isikan terminology yang sesuai untuk definisi dibawah ini:

[Instance variables] The memory that the instance of the class has. That memory can be saved and accessed later as long as the instance exists.

[Variable] A field used to store information about the class to use immediately or later.

[Parameter] Used to pass values to methods to specify how objects are to move, or to tell objects what type of response we expect when we ask objects a question.

[Inheritance] Each subclass can use (inherits) the methods from its superclass.

[Return type] Word at the beginning of the method that tells us what type of information a method call will return.

[Methods] Set of operations or tasks that instances of a class can perform. When a method is invoked, it will perform the operation or task specified in the source code.

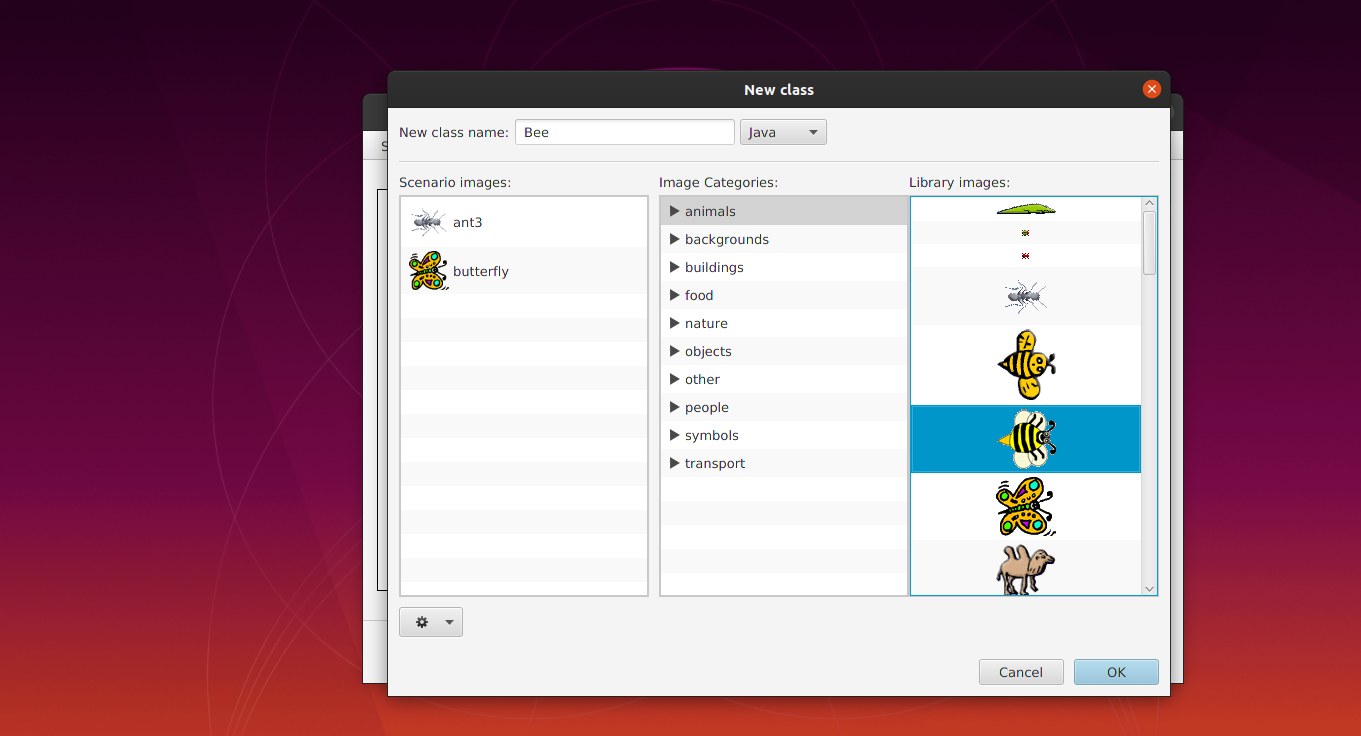
[Method call] A method call instructs the instance to perform an operation or task. You can read the method to understand what operation or task is to be performed.

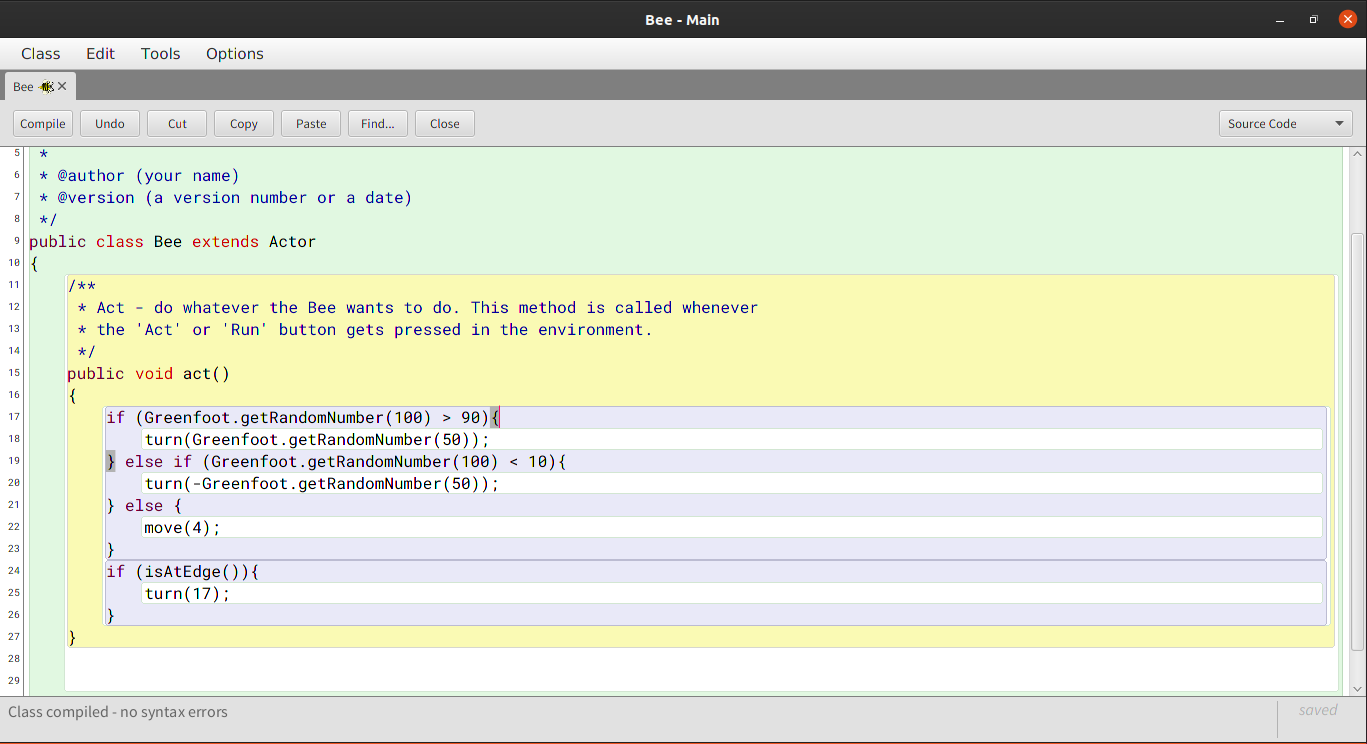
[Method signature] Access, return type, name, and parameters for a method.

[Debugging] Process of finding and removing bugs—or errors—in a computer program.

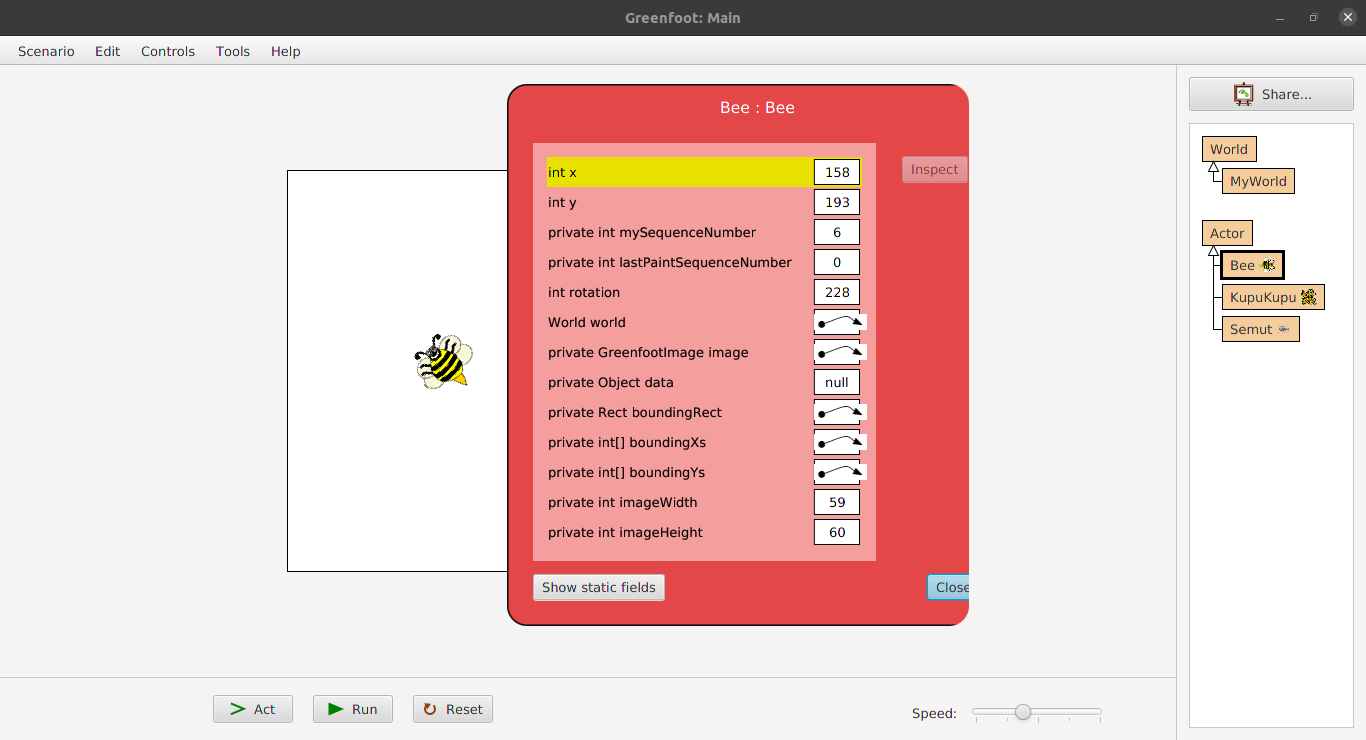
**TRY IT / SOLVE IT:**

1. Buka aplikasi Greenfoot, Periksa daftar method yang dimiliki oleh class actor & sebutkan setidaknya tiga method yang berbeda dari class actor. Jelaskan bagaimana masing – masing parameter mereka digunakan.
   * GreenfootImage getImage()
     + Method ini dipanggil tanpa menggunakan parameter. Method ini berfungsi untuk mendapatkan nilai return berupa jendela inspect yang berisi method – method yang digunakan untuk memroses gambar dari objek yang kita buat
   * boolean isAtEdge()
     + Method ini dipanggil tanpa menggunakan parameter. Method ini berfungsi untuk memberitahu kita bahwa objek yang kita buat telah menyentuh batas skenario. Jika sudah menyentuh batas, maka nilai true akan dikembalikan. Sebaliknya nilai false yang akan dikembalikan.
   * void setLocation(int, int)
     + Method ini dipanggil dengan menggunakan dua buah parameter. Method ini berfungsi untuk mengatur lokasi dari objek yang telah kita buat.
2. Buat scenario baru didalam greenfoot, kemudian buat kelas Bee sebagai subclass dari class actor. Tambahkan kode program di kelas Bee untuk memindahkan (move) dan membelokkan (turn) object dengan nilai parameter yang Anda tentukan. Tambahkan objek Bee kedalam workd/scenario kemudian jalankan skenario untuk melihat bagaimana parameter yang Anda masukkan dalam metode act () mempengaruhi gerakan instance.

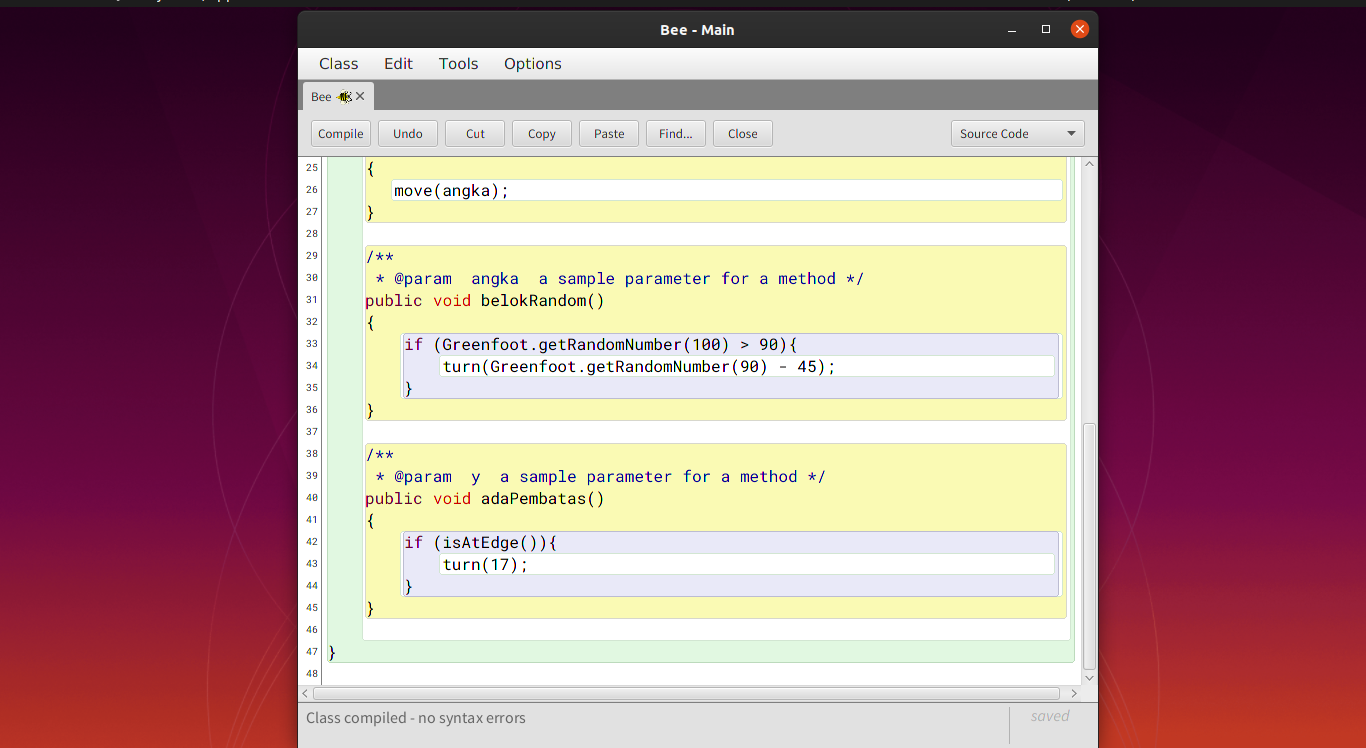
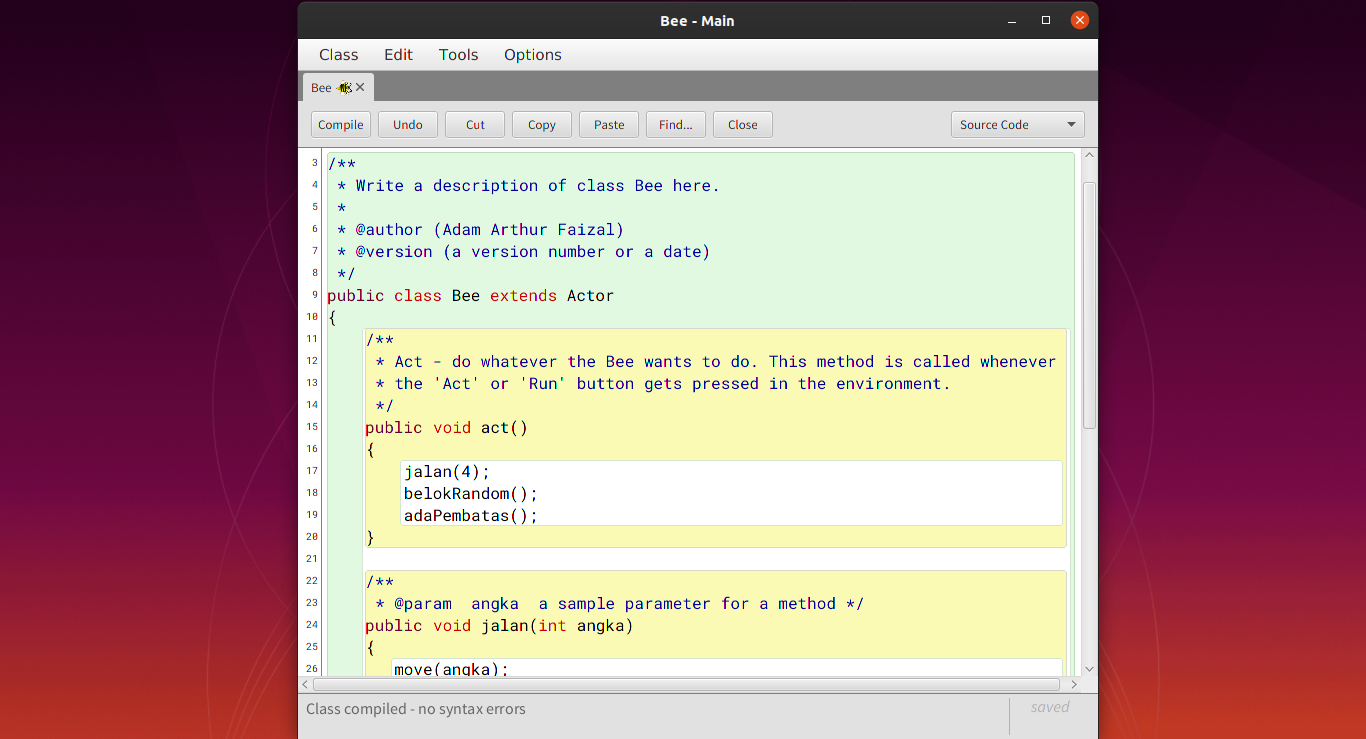


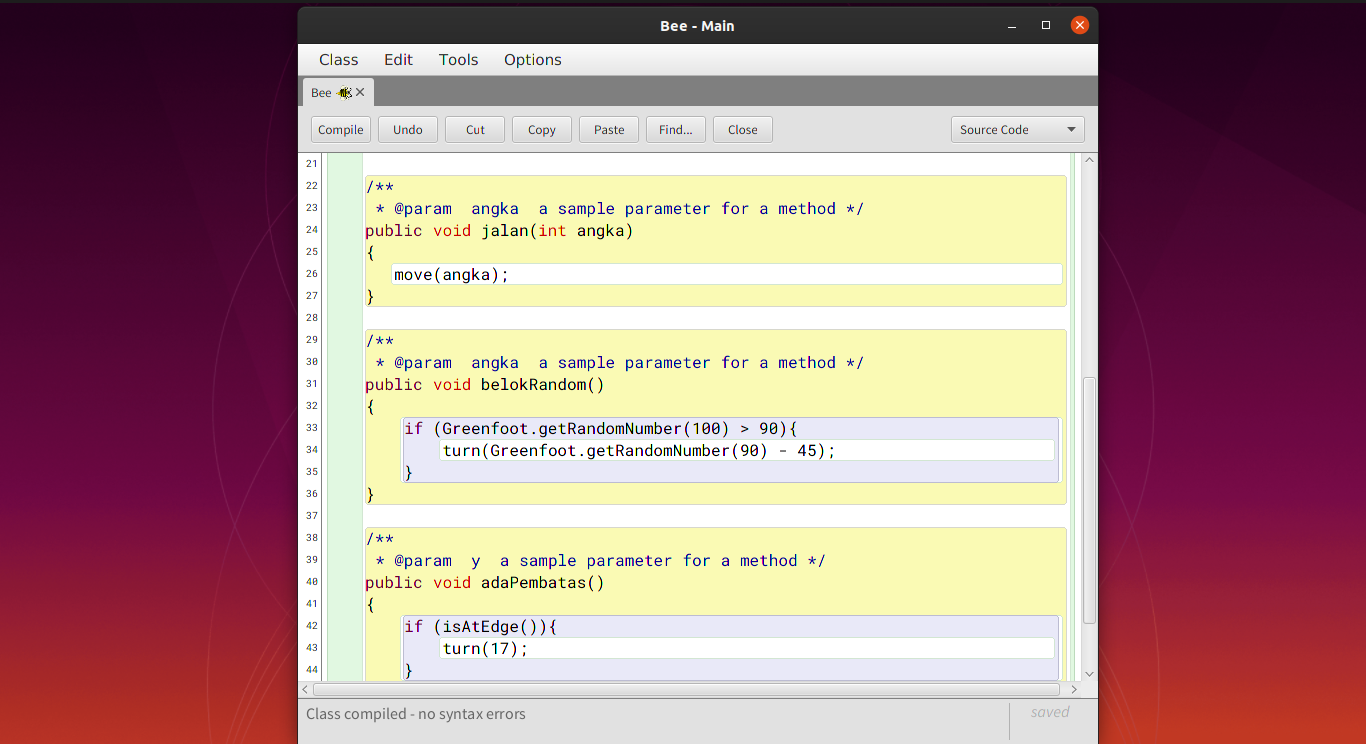


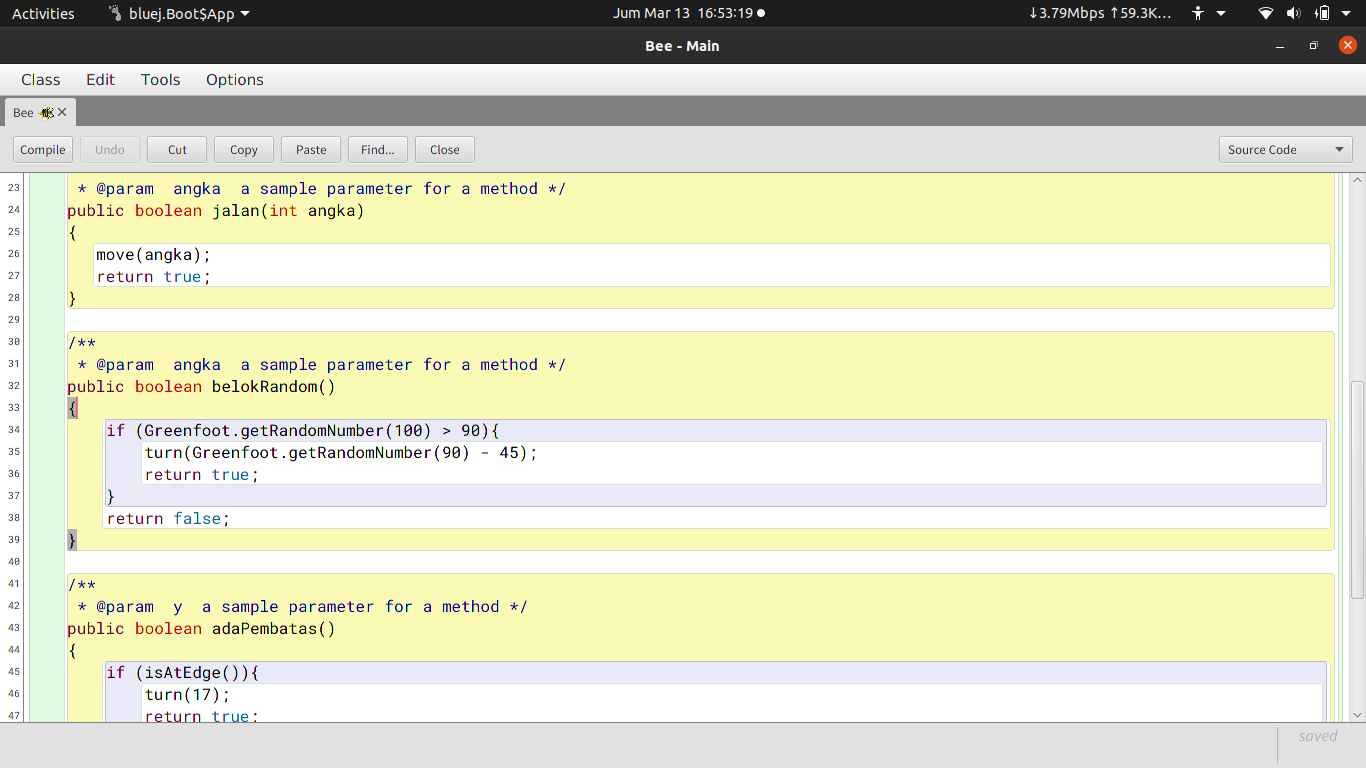
1. Periksa property dari instance/objek yang Anda tambahkan ke dalam skenario. Catat nilai variabel lokasi objek (koordinat x dan y) saat ini.

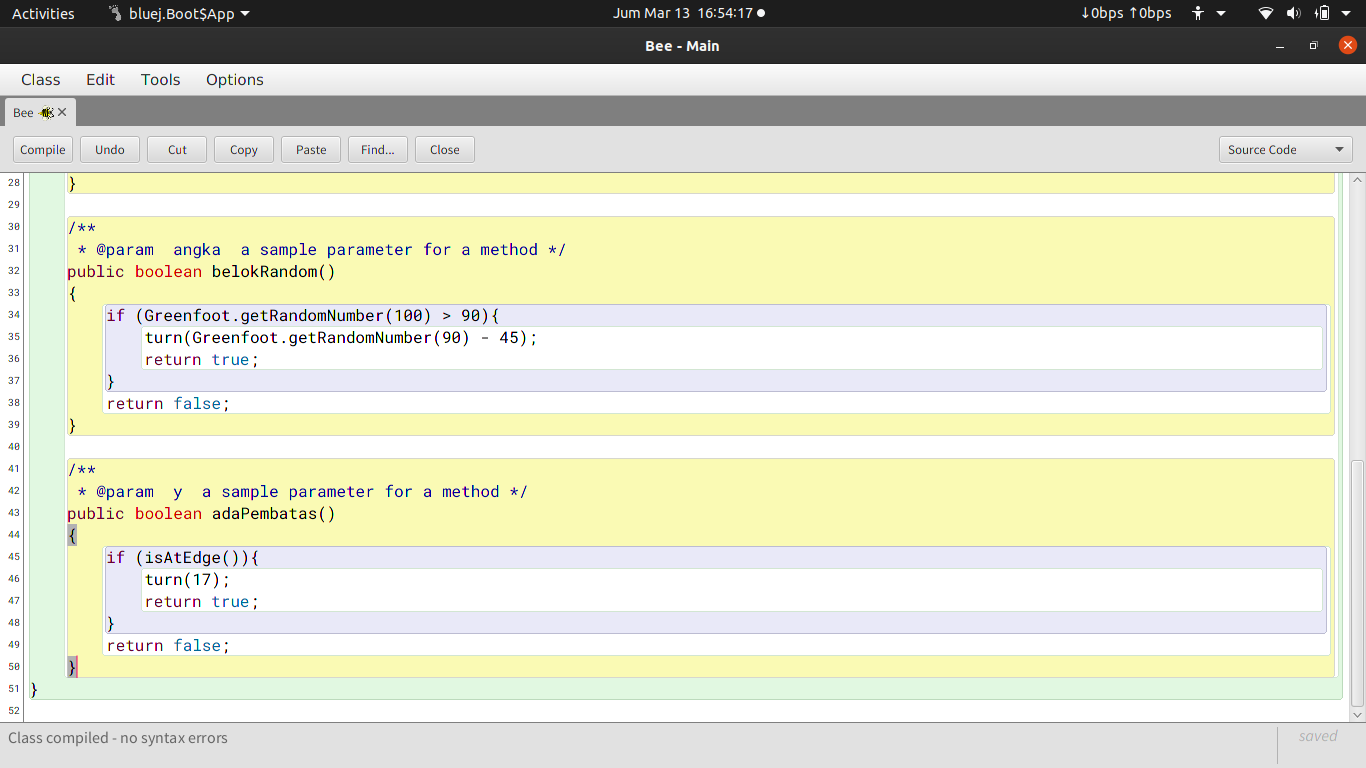


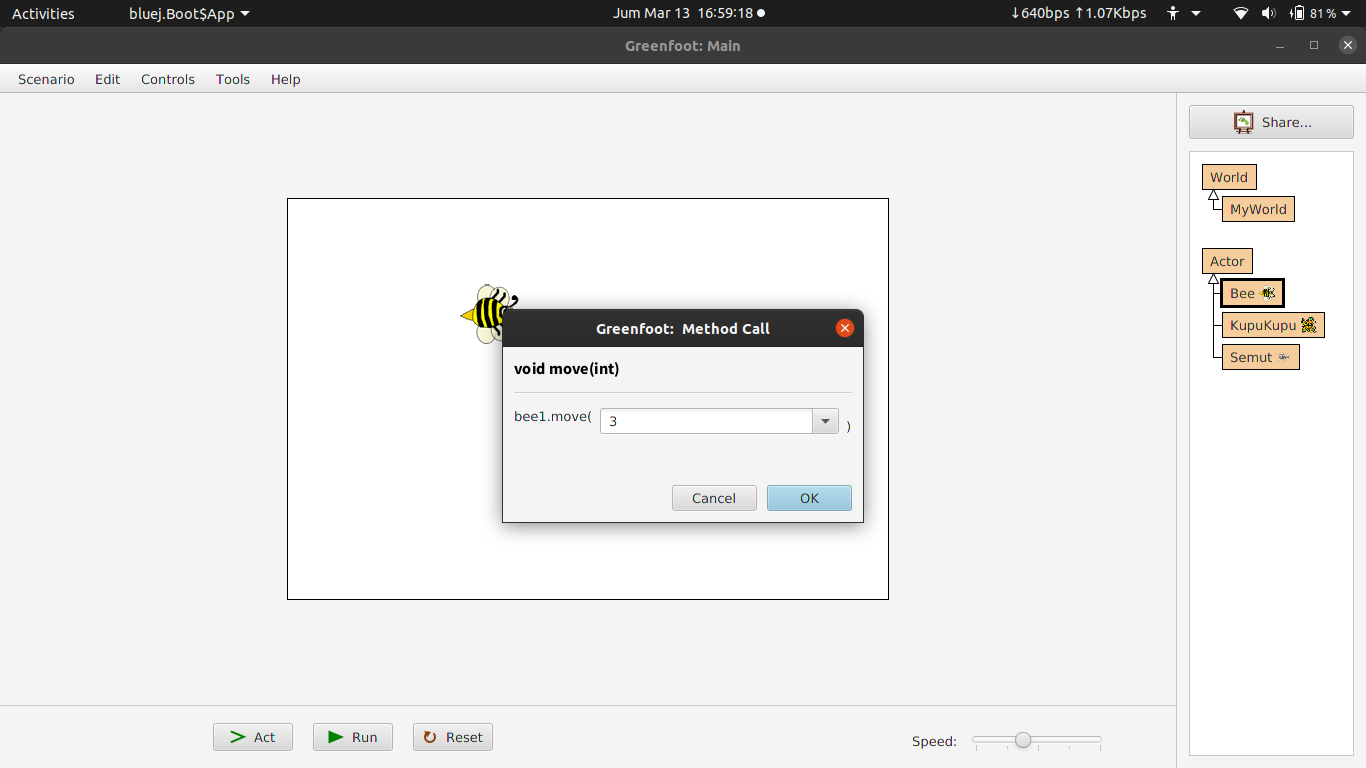
1. Jelaskan perbedaan antara method dengan void dan non-void return type. Tambahkan setidaknya 3 method yang berbeda dengan jenis pengembalian void dan non-void kedalam kelas Bee. Jalankan skenario untuk melihat hasil penambahan method.
   * Method dengan return type void adalah method yang tidak memiliki return, jadi method ini tidak mengembalikan nilai apapun

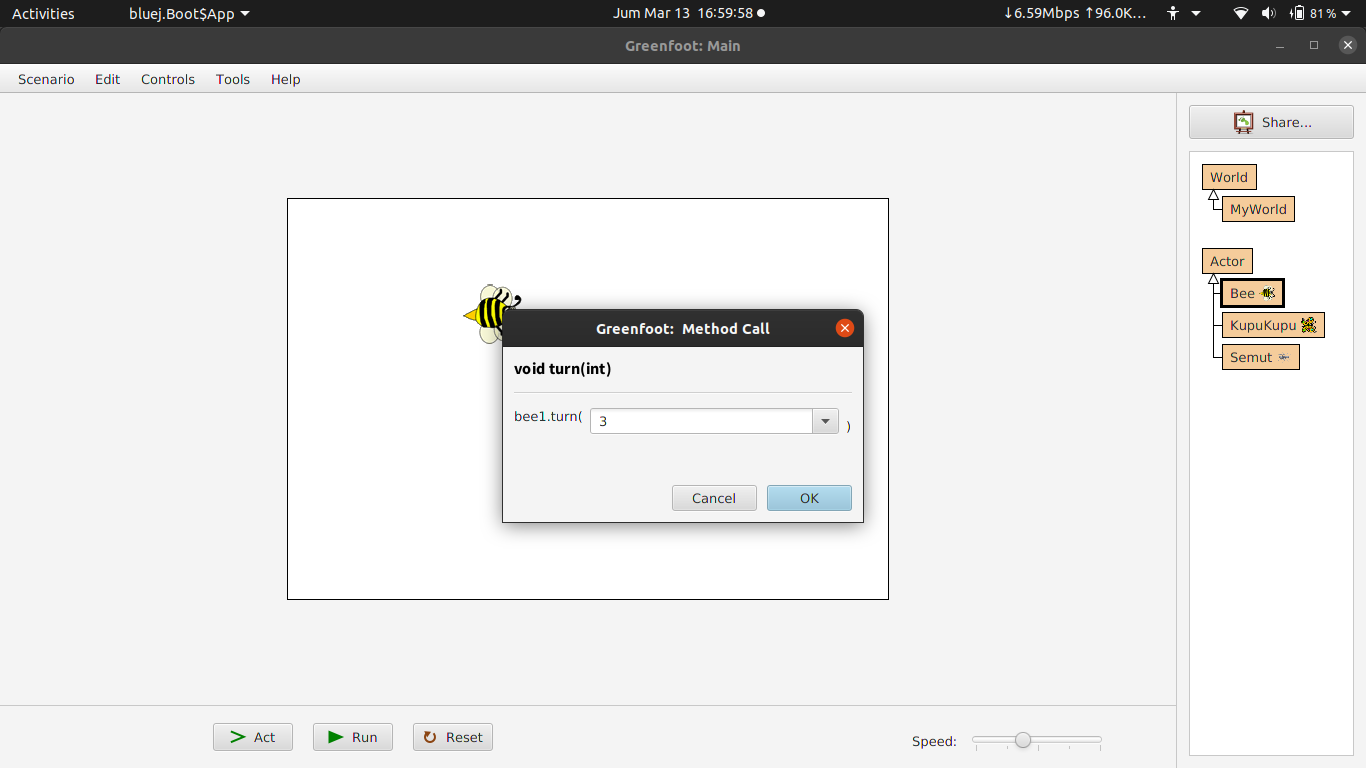


* + Method dengan return type non-void adalah method yang memiliki return. Method ini mengembalikan nilai tertentu

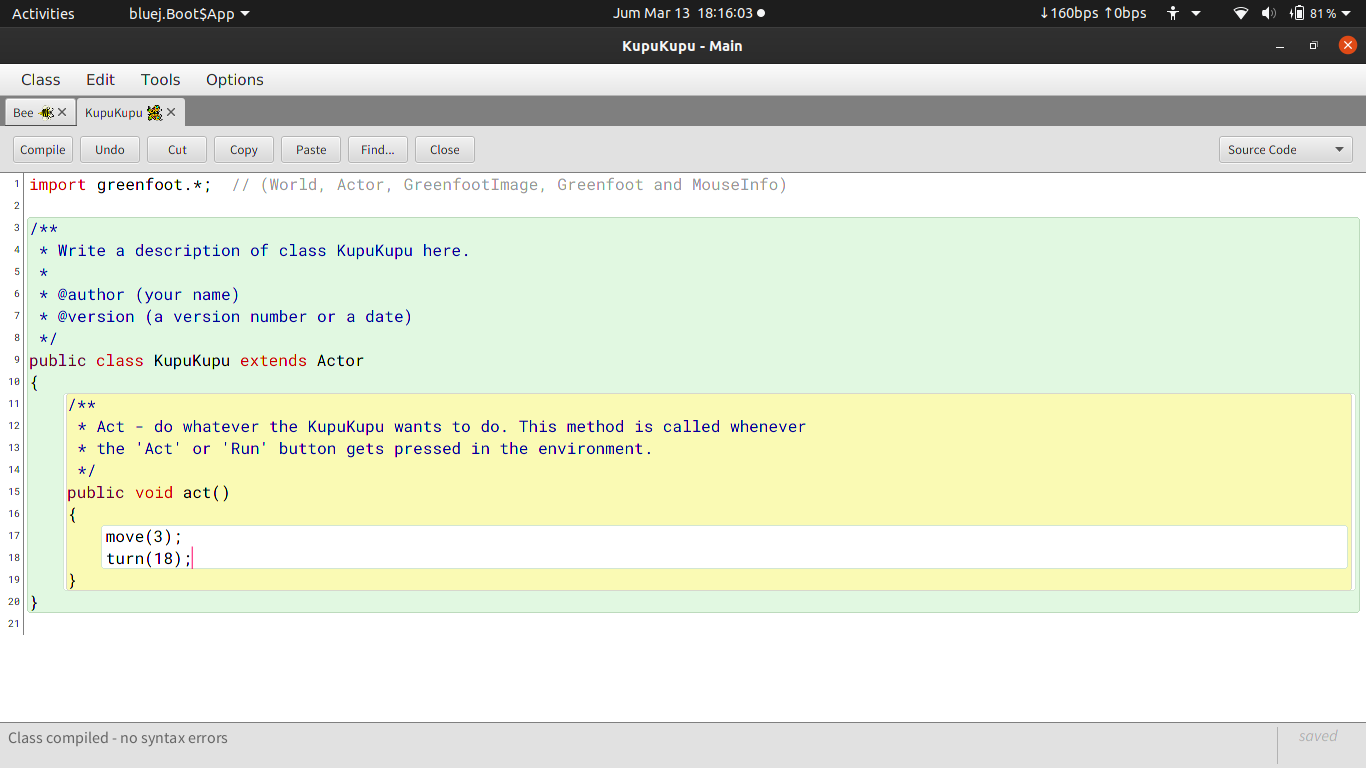


1. Tambahkan objek ke skenario, dan kemudian memanggil dua metode yang berbeda yang memerlukan parameter. Tuliskan *method signature* nya, dan apa yang terjadi pada instance setelah metode dipanggil.



* + void move (int)
    - void sebagai return type void atau tanpa nilai kembalian
    - move adalah nama method
    - (int) adalah parameter dari method
  + void turn (int)
    - void sebagai return type void atau tanpa nilai kembalian
    - turn adalah nama method
    - (int) adalah parameter dari method

1. Dalam skenario Anda, Edit kode program kelas Bee agar objek Bee berpindah 3 langkah, dan kemudian berbelok 18 derajat ke arah kiri. Kompilasi kode, dan kemudian uji hasil program Anda.



1. Benar atau salah: list parameter kosong dari sebuah method akan memerlukan nilai/data untuk memanggil method ini.
   * Salah, karena:
     + Jika suatu method tersebut tidak memiliki parameter , maka untuk memanggil method itu tidak perlu menggunakan nilai parameter)

***Setelah sesi praktikum SELESAI, laporan praktikum harus dikirim/diupload ke google classroom pada hari yang sama.***