**MODUL 4. Randomize, Defining Method, Collision Detection**

**Capaian Pembelajaran Praktikum:**

* Membuat behavior random
* Mendefinisikan method baru
* Menerapkan collision detection di Greenfoot
* Membuat scenario interaktif dengan keyboard dan sound

**Tools:**

* Java Development Kit (JDK)
* Greenfoot IDE

**Terminologi:**

Isikan terminology yang sesuai untuk definisi dibawah ini:

[Defined Methods] A new method that a class didn't already possess; these methods are written in the class's source code below the act() method.

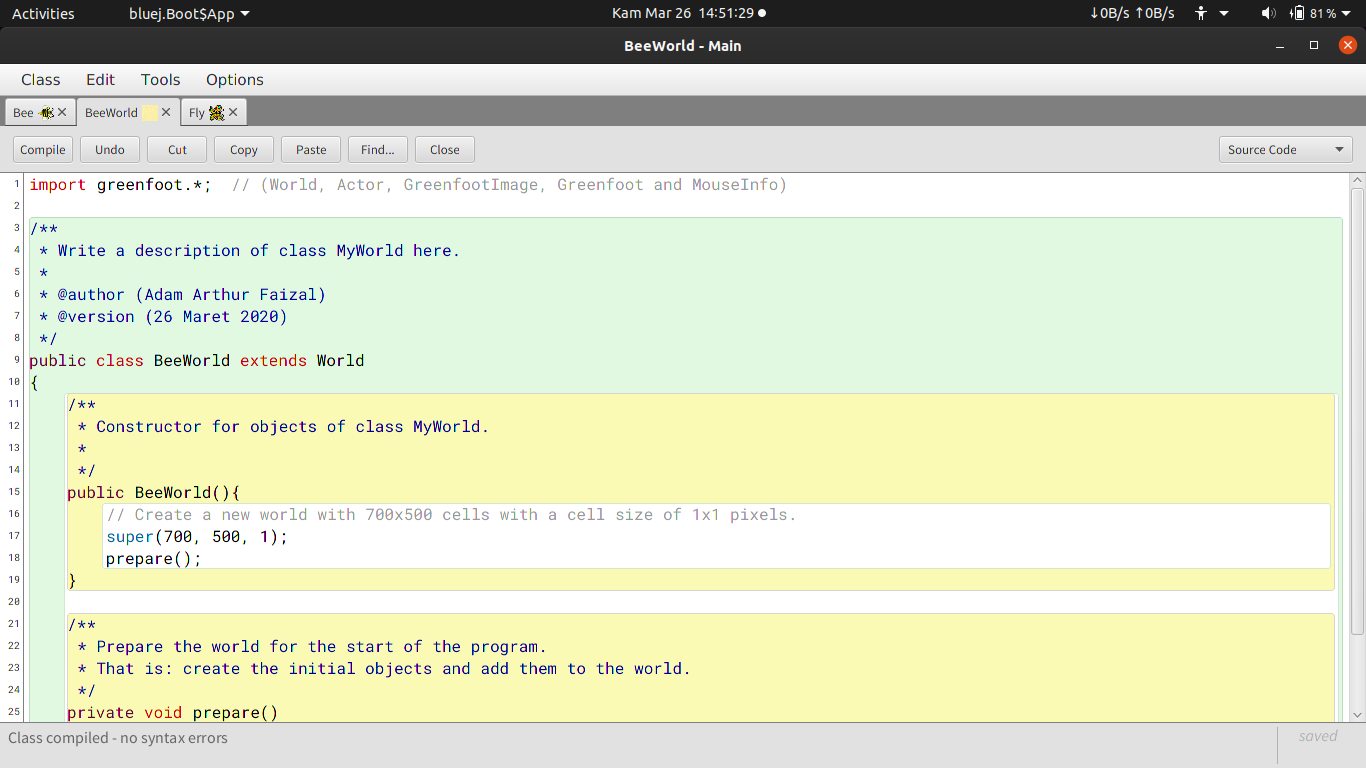
[Dot Notation] A technique that allows a class to use a method from another class or object. The dot between the class/object name and the method name indicates that the method comes from a different class or object.

[Constructor] A special kind of method that is automatically executed whenever a new instance of the class is created.

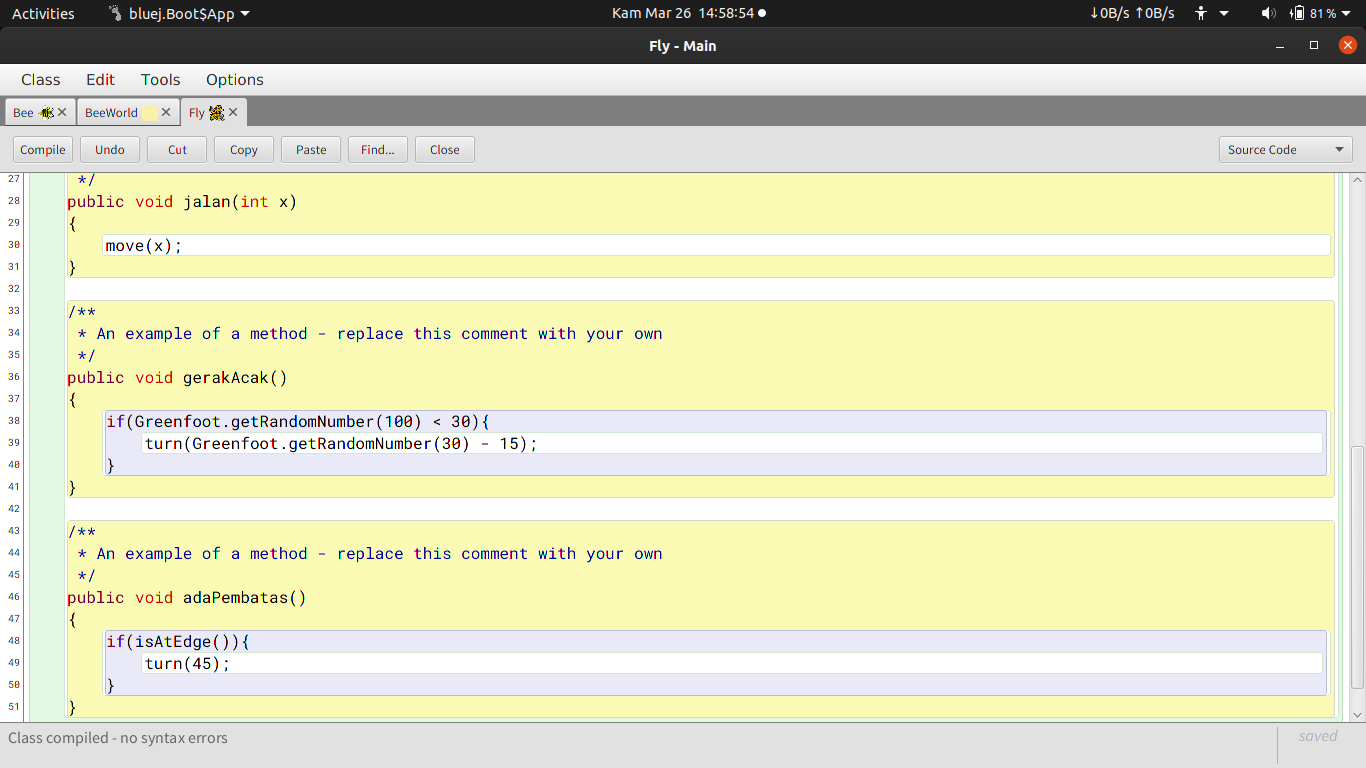
[new Keyword] A keyword that indicates that a new object is being created.

**TRY IT / SOLVE IT:**

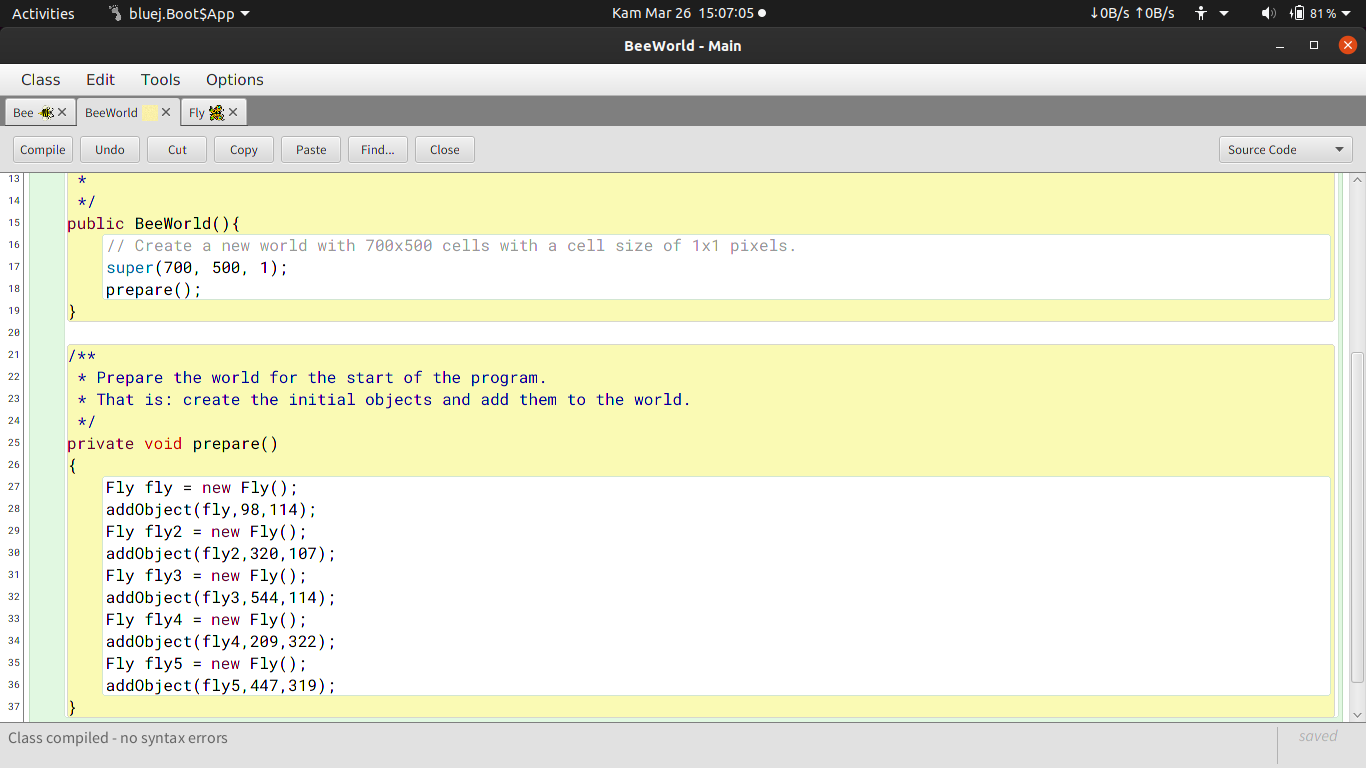
1. Buat sebuah scenario baru. Buat subclass BeeWorld sebagai background dari scenario dengan ukuran lebar 700, tinggi 500 dan resolusi 1. (Gunakan image sand.jpg dari kategori backgrounds)



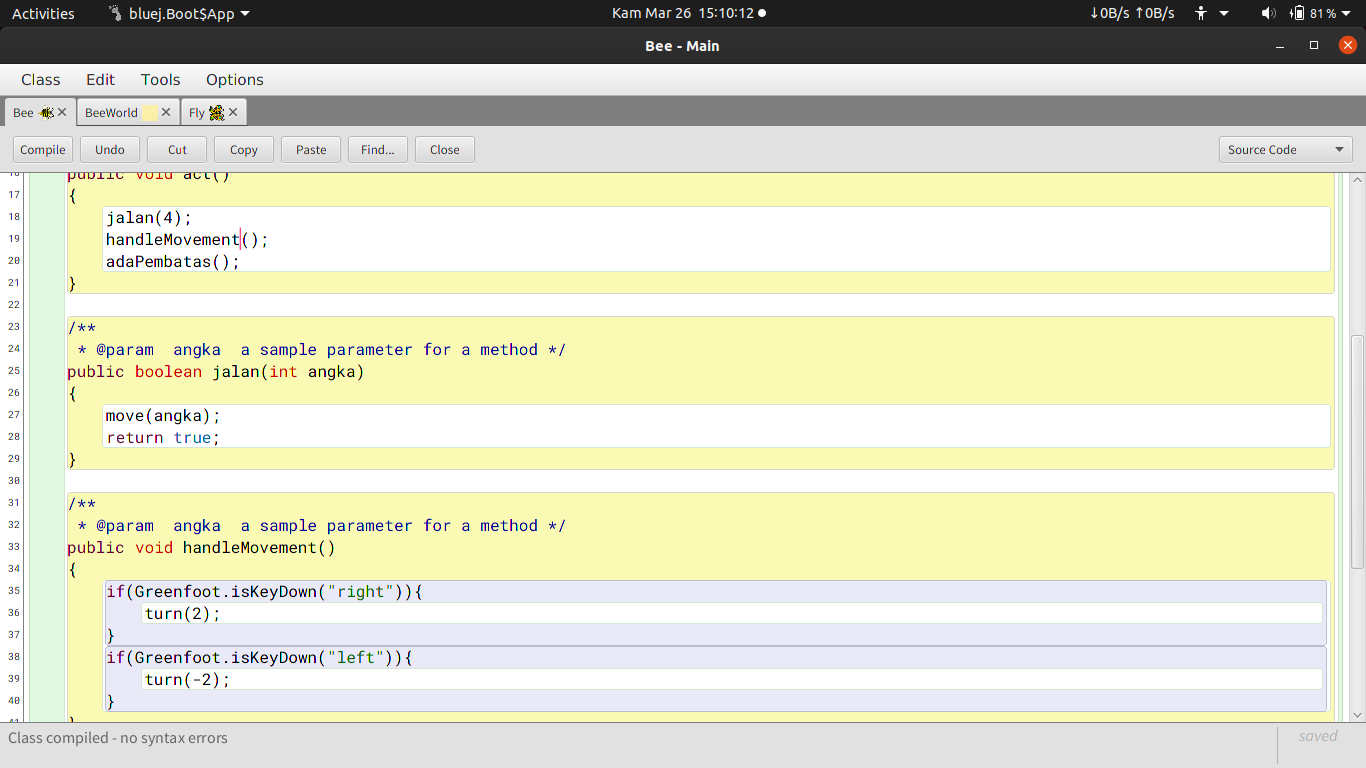
1. Tambahkan class Fly sebagai subclass Actor. Buatlah sebuah method *gerakAcak()* untuk Fly agar bergerak secara acak mengikuti kaidah sbb: Fly akan berbelok secara acak sampai dengan 15 derajat, dengan probabilitas 30%. Kemudian invoke atau call method *gerakAcak()* di method *act().* (Gunakan image fly.png dari kategori animals)



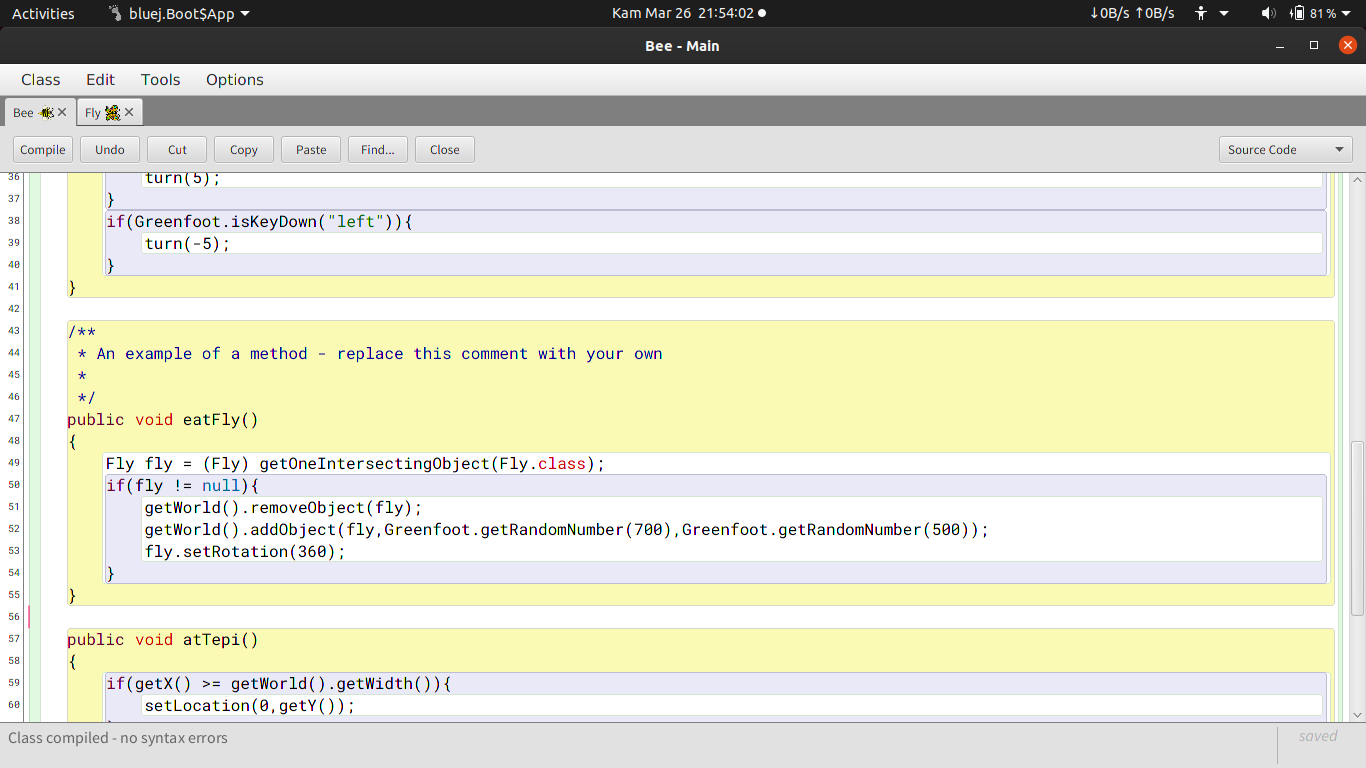
1. Dalam scenario yang anda buat, tambahkan 5 objek Fly kedalam BeeWorld secara otomatis dengan lokasi penempatan dibuat acak baik untuk koordinat X dan koordinat Y. (Gunakan method addObject untuk menambah objek & tambahkan kedalam konstruktor BeeWorld)



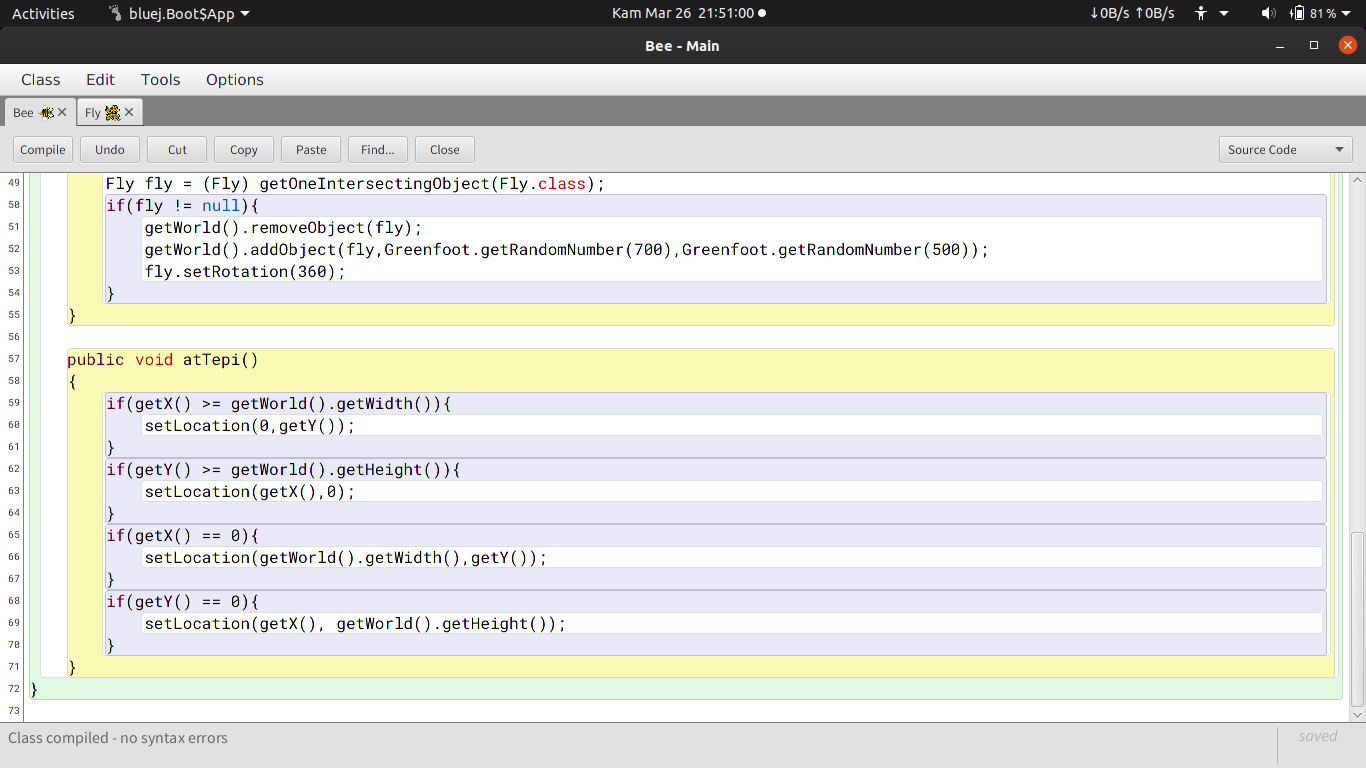
1. Buat class Bee dan buat method *handleMovement()* untuk menangani pergerakan Bee, yaitu Bee akan bergerak maju dengan kecepatan 2 dan akan berbelok kekiri sebesar 2 derajat ketika ada tombol left dari keyboard ditekan. Dan begitu pula sebaliknya, Bee akan berbelok ke kanan sebesar 2 derajat ketika tombol right dari keyboard ditekan. (Gunakan method isKeyDown dan statemen if)



1. Buat method *eatFly()* di class Bee untuk menerapkan collision detection. Jika Bee berada dilokasi yang sama dengan objek Fly, maka objek Fly akan hilang / di remove dari BeeWorld. Kemudian objek Fly baru akan ditambahkan kedalam BeeWorld secara acak dengan jarak minimal 10 dari tepi. Panggil method yang anda buat di method act().



1. Tambahkan method *atTepi()* di class Bee, yaitu ketika objek Bee berada di tepi kanan maka Bee akan berpindah ke tepi sebelah kiri. Jika Objek Bee berada di tepi atas maka objek Bee akan berpindah ke tepi bawah. Begitu pula sebaliknya.



***Setelah sesi praktikum SELESAI, laporan praktikum dan source code (zip) harus dikirim/diupload ke google classroom pada hari yang sama.***