|  |  |  |
| --- | --- | --- |
| **Python. Test lesson #3** | | |
| **Student:** | **How many lessons passed:** | **Date:** |
| **Profile link:** | | |

|  |  |  |
| --- | --- | --- |
| **Theory**(oral survey, 1 -2 question from the chapter, in case of difficulty, you can ask another question from the chapter) | | Mark  +/- |
| **Topic 1 Classes** (Covered in lessons: 15, 16, 16.1) | |  |
| What are the classes in Python? | This is the description of the object, its behavior, what can it do, and what options does it have |  |
| How to create a Class? | Type the "class" command and name it |  |
| What is the object of the class? | This is the instance of the object. The object gets all the options and methods of the class |  |
| How to create the instance of the class? Is it necessary to type something in the brackets in the instance of the class? | Create the variable and call the class inside  We should type something in the brackets if it's necessary to set the characteristics of the object |  |
| What is the method of the class? | This is the function which the class has. The ability of the class object |  |
| How to call the class method? | To call the method, you need to write something like "object.method()." |  |
| What is the method which is used when the class is created? (We write this method in properties of the object) | The "\_\_init\_\_" command |  |
| What do we usually write as the first argument for every object method? | The "self" command |  |
| **Topic 2 Inheritance** (Covered in lessons: 15, 16, 16.1) | |  |
| What is the inheritance of the class? | Getting attributes (properties) and methods from the main class |  |
| How to inherit the class? | Add the name of the parent class to the child class |  |
| How can the child class initialize (get) the same properties from the parent class? | Using the “super().\_\_init\_\_” command |  |
| **Total** | |  |

|  |  |  |
| --- | --- | --- |
| **Practice**.  Note :   * mark (2) is put if the student independently invented and implemented the solution, * mark (1) is set if the solution was invented or implemented with a hint, or a hint was provided when the error was eliminated * the mark (0) is set if the solution was invented and implemented with a hint, or a hint was provided when fixing an error | | Mark  0..2 |
| The attributes of the class Insect is set. | The variables for keeping the object's name, amount of food, and satiety are created. The "name" in the brackets is set. |  |
| The decrease of food amount while running the "eat" method and the increment of satiety is created | Student decreased a "self.food" ( or another name) by 1 and increased a "self.hunger" (or another name) by 1 |  |
| If there is no food, the insect can't eat | The condition of the ability of eating is used |  |
| The information about the food amount, satiety, ability of eating is printed | The "else" condition is added. If the condition is true, the information about food amount, satiety, and the insect's eating will appear. Otherwise, the information about food amount, satiety and nothing to eat will appear |  |
| The method of searching for food is created | The "self.food" is increased, the "self.hunger" is decreased, the information about the actions, amount of food, and satiety is shown. |  |
| The child class "Bee" is created | The name of the parent class is in the brackets of the child class "Bee." The properties are inherited using the "super()" command in the "\_\_init\_\_" method. |  |
| The attribute of food amount is added | The "self.honey" is created in the "\_\_init\_\_" and assigned the initial integer |  |
| Instance object of the "Bee" class is created | The object bee is created and assigned with the name of the class Bee and given the name of the bee |  |
| The method of collecting the honey | The satiety should be decreased while collecting the honey, and at the same time, the amount of honey should be increased |  |
| The output information about the condition of the bee condition is met. | The information is shown while running the method. The bee has collected the honey, the amount of food, satiety, and amount of honey |  |
| The method of bee living "live" is run | The "live" method is applied to the instance object of the Bee class after creating the instance object |  |
| The life duration of the bee is performed for 30 days | The loop is repeating 30 times and calling the "live" method |  |
| The number of the day information is performed | There is a "print" function in the loop where the iterator is used |  |
| The breaking the loop is performed if the bee is dead | The "living" variable from the "live" method is used, and all the values are checked using the if condition. If the condition is "False," the loop will end |  |
| The work of the "live" method is explained | The method reports on the death of the bee. Also, the method tries to react to satiety on time (If it's lower than some value, the bee must eat). In addition, the method checks, if there is no food to eat, the bee must search for the food. Otherwise, the bee chooses randomly what to do. |  |
| **Total** | |  |

**Practical notes :**

**Assessment of student's attitude to learning (on a scale of 1 to 5):**

Interest -

Motivation -

Engagement -

Academic achievement –

***During the lesson, the student showed himself as (choose 2-3):***

* Attentive
* Plodding
* Curious
* Hardworking
* Creative, innovative
* Inventive
* Your option

***Qualities requiring further attention and development (choose 2-3 ):***

* Mindfulness,
* Perseverance
* Hard work
* Homework
* Mathematical calculations
* Ability to present, tell and explain logically
* Creation
* Initiative

***Topics of particular interest to the student:***

* Programming
* Painting
* The gameplay itself

***Recommendations from the teacher:***

* Continue course
* Change course
* Increase the number of independent studies to….
* Reduce the number of independent studies to….