|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **#** | **Lecture Topics** | **Lab Content** | **Story: to do** | **Notes** |
| 1 | Introduction (Motivation), history of python, Strings, Numbers, Loops, Functions, List, Tuple, Dictionary | The TA shall start with the students to explain the target, which is to simulate the story of the rabbit.  Then Students will have to define variables for each animal questions, for example:  #dog:  dog\_sports\_questions = “jump 100 times”  dog\_food = “I eat {0} bones and {1} birds”  dog\_number\_bones\_food = 5  dog\_number\_birds\_food = 5  etc...  Then after defining all the variables related to each animal, they will write a function for the story scenario, which will interact with the user to press enter to continue. The story scenario will start with the rabbit with his mother, and then loops on 5 animals and their requests, and ends by the rabbit at his house. | Full Scenario of the story, with 5 characters, each asking a question to be done by the rabbit.  Explanation:  - **List** of animals that the rabbit will meet  - **Function** for talking (accepts a string to print)  - **Function** to make operations (accepts 2 operands and an operation, contains **if conditions**)  - Function to return **Sum, AVG, Min, Max** of a given a list of numbers using the built-in methods  Mother Rabbit:  - Say Good Morning: **string**  - Prepare the food for the rabbit: number of carrots and number of lettuce **numbers**  Rabbit:  - say everyday carrots and lettuce 5 times! **Loop**  **-** say “I am leaving”, “mom”, **string concatenation**  Dog:  - dogFood: “I eat” 5 ” bones and ” 6 “birds”  **integer to string/ string concatenations/ string.format**  - Ask to jump 100 times, **loops**  Cat:  -Singing: **Loops**  -Indexing from end in lists l**ist [-1], list[-3]**  -**string Slicing**  -”You you can take take take only one one one fish fish” **Sets** ”You can take only one fish”  Fox:  -Ask the rabbit to call his mom for a fish **Dictionary**  - Use maths package operations **factorial, sqrt**  -Length of 3\*Python **len(constant \* str)**  - **Reverse, append, sort** list of int, strings  Monkey:  -Say jokes:**strings**  -Ask smart questions like mathematical operations and Matrix operations.  -Food: Banana  Lion:  -Loud Shouts: **Strings To upper “I AM THE KING”**  -Ask many questions about rabbit: **Tuples**  -Food: Meat  - rabbit answers based on the answers he knows in his answer\_Dictionary, otherwise says I am not sure!  - Count the number of success questions, otherwise the lion will eat the rabbit, the rabbit will have then to run back to his house. | Console Applications  Perspective: Problem Solving  TA: Nourhanne |
| 2 | - Object Oriented Programming.  - Importing packages | Class Story\_Character.  Data Members:  Character\_Name: **String**  Character\_Words : **String / List of Strings.**  Character\_Food: **String**  Character\_Questions : **Set of string.**  Methods:  Say\_CharacterWords().  Eat().  Run().  Say\_Questions().  Create Object for each Character. | -HighLight the main concepts of OOP.  -Link OOP concept with our story.  -HighLight the available classes for the story (Data Members and Methods).  -Import some packages for illustrations (Math library etc...). | OOP:   * Encapsulation * Access modifiers * Classes * Objects * Constructors, Destructors, * Isinstance   TA: Nouran |
| 3 | - PyGame |  |  | TA: Bassel |
| 4 | - Exceptions  - Saving and restoring from files. | -Write function to load questions based on game level.  -Load\_Game\_Questions(int Level):  -Conditions based on level value.  -Read Questions and answers from same file and tokenize based on delimiters.  -Add Questions to Questions\_List.  -Add Answers to Answers\_List. | -Link between files and exceptions.  -Use files to load questions and answers from files depending to game level. | Exception:  Cover only Built-in exceptions  TA: Nouran & Nourhanne |
| 5 | - More Language Features (Lambdas,  Maps, Zip, Generators) |  | Apply examples on each of these concept and then adjust enhance parts of the code based on the newly learned concepts, also prepare to add new requirements in the story if no change to be made | This lecture is canceled |