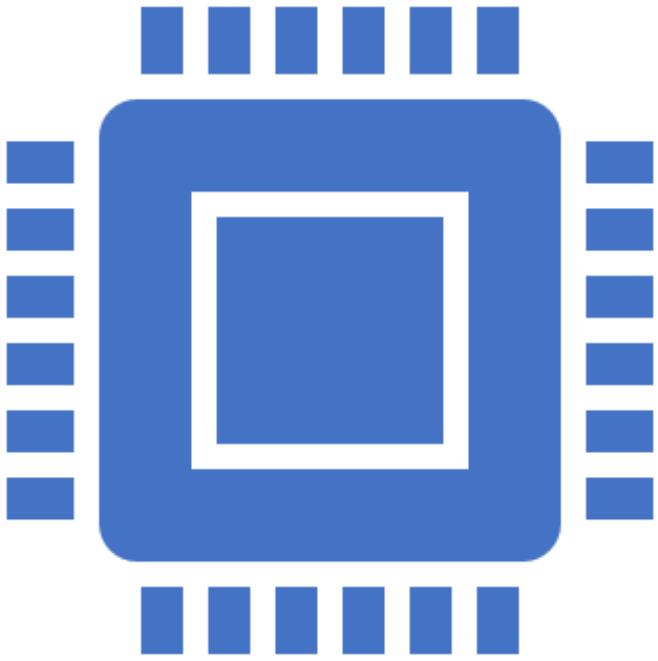


```
mirror_mod = modifier_ob.  
set mirror object to mirror.  
mirror_mod.mirror_object  
operation == "MIRROR_X":  
mirror_mod.use_x = True  
mirror_mod.use_y = False  
mirror_mod.use_z = False  
operation == "MIRROR_Y":  
mirror_mod.use_x = False  
mirror_mod.use_y = True  
mirror_mod.use_z = False  
operation == "MIRROR_Z":  
mirror_mod.use_x = False  
mirror_mod.use_y = False  
mirror_mod.use_z = True  
  
selection at the end -add  
mirror_ob.select= 1  
modifier_ob.select=1  
context.scene.objects.active  
("Selected" + str(modifier_ob.  
mirror_ob.select = 0  
= bpy.context.selected_object  
data.objects[one.name].select  
  
print("please select exactly  
  
-- OPERATOR CLASSES ----  
  
types.Operator):  
X mirror to the selected  
object.mirror_mirror_x"  
mirror X"  
  
context):  
context.active_object is not
```

Software Testing

| Foundation Certification Level
| By: Anuruddha Abeysinghe
| SLIIT Academy



What it is?

- It is the process used to identify the correctness, completeness and quality of developed computer software.
- Executes the program/application under positive and negative conditions by manual or automated manner.
- It mainly checks for the
 - **Specification**
 - **Functionality**
 - **Performance**

Why we need a software testing?

- To discover the defects/bugs before the delivery to the client, which guarantees the quality of the software.
- To makes the software more reliable and easy to use.
- To ensure the reliable and high-performance software operation.
- To ensure about the security of the system.
- To enhance the customer satisfaction.



- **Error** – Human action that produces the incorrect result that produce a fault.
- **Bug** – The presence of error at the time of the software.
- **Fault** – State of the software caused by an error.
- **Failure** – Deviation of the software from it's expected result. It is an event.

Software Testing Methods

- ***Black – Box Testing***

- Functionalities of software applications are tested without having knowledge of internal code structure, implementation details and internal paths.
- Mainly focuses on input and output of software applications and it is entirely based on software requirements and specification

- ***White – Box Testing***

- Internal structure, design and coding of software are tested to verify flow of input-output and to improve design, usability and security.
- Tester knows the internal process, paths and code structure.

Software Testing Levels

- **Unit Testing**

Find More - <https://www.guru99.com/unit-testing-guide.html>
https://www.youtube.com/watch?v=lj5nnGa_Dlw&t=53s

- **Integration Testing**

Find More - <https://www.guru99.com/integration-testing.html>
<https://www.youtube.com/watch?v=QYCaaNz8emY>

- **System Testing**

Find More - <https://www.guru99.com/system-testing.html>
<https://www.youtube.com/watch?v=N8-qNMHOVyw>

- **Acceptance Testing**

Find More - <https://www.guru99.com/user-acceptance-testing.html>
<https://www.youtube.com/watch?v=N8-qNMHOVyw>

THANK YOU

