

INFORMATION SYSTEMS 3A

INTRODUCTION TO SOFTWARE ENGINEERING

Contents

- Introduction of Software Engineering
- Need of Software Engineering
- Characteristics of Good Software
- General issues that affect software
- Nature of Software Engineering
- Software engineering fundamentals
- Changing Nature of Software
- Software engineering ethics
- Software Process Activities
- SDLC Models



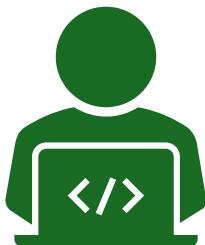
The image shows a person's hand holding a smartphone. The screen of the phone displays a block of Python code. The code is related to Blender's mirror modifier, specifically for the 'MIRROR' operator. It includes logic to set up the mirror object, handle different mirror operations (X, Y, Z), and manage selection of objects. The code uses various Blender API functions like `bpy.context.scene.objects.active`, `bpy.context.selected_objects`, and `bpy.types.Operator`. The background of the phone screen is dark, and the text is in a light color.

```
mirror_mod = modifier_obj
# Set mirror object to mirror
mirror_mod.mirror_object = mirror_object
if operation == "MIRROR_X":
    mirror_mod.use_x = True
    mirror_mod.use_y = False
    mirror_mod.use_z = False
elif operation == "MIRROR_Y":
    mirror_mod.use_x = False
    mirror_mod.use_y = True
    mirror_mod.use_z = False
elif 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
#other_ob.select=1
context.scene.objects.active = bpy.context.active_object
("Selected" + str(modifier))
mirror_ob.select = 0
bpy.context.selected_objects.append(mirror_ob)
data.objects[one.name].select = 1
print("please select exactly one object")
#OPERATOR CLASSES -----
#types.Operator:
#    X mirror to the selected object.mirror_mirror_x"
#    or X"
#context):
#    context.active_object is not None
```

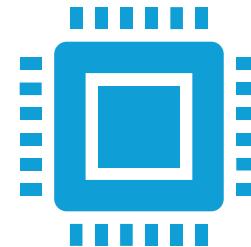
What Is Software?



The software is collection of Integrated programs



Software consists of carefully organized instructions and code written by programmers in any of various special computer languages.



Computer programs and associated documentation such as requirements, design models and user manuals.

What Is Engineering?

- Engineering is the application of scientific and practical knowledge in order to invent, design, build, maintain, and improve systems, processes, etc.

What is software Engineering?

- Software engineering is an engineering discipline that is concerned with all aspects of software production.
- According to IEEE's definition software engineering can be defined as the application of a systematic, disciplined, quantifiable approach to the development, operation, and maintenance of software, and the study of these approaches; that is, the application of engineering to software.

Need of Software Engineering

- **Handling Big Projects:** Corporation must use SE to handle large projects without any issues.
- **To manage the cost:** Software engineering programmers plan everything and reduce all those things that are not required.
- **To decrease time:** It will save a lot of time if you are developing software using a software engineering technique.
- **Reliable software:** It is the company's responsibility to deliver software products on schedule and to address any defects that may exist.
- **Effectiveness:** Effectiveness results from things being created in accordance with the software standards.

Characteristics of Good Software

Operational	Transitional	Maintenance
Budget	Interoperability	Flexibility
Efficiency	Reusability	Maintainability
Usability	Portability	Modularity
Dependability	Adaptability	Scalability
Correctness		
Functionality		
Safety		
Security		

General issues that affect software

- Heterogeneity
- Business and social change
- Security and trust
- Scale

Software products

Software:

- “Computing systems and documentation. Software products may be developed for a particular customer (custom) or may be developed for a general market (generic)”

Software engineering:

- “Software engineering is an engineering discipline that is concerned with all aspects of software production

Nature of Software

1. System Software: System software is software designed to provide a platform for other software's.
 - It is a interaction between hardware & application software.
 - Examples: Operating Systems like macOS, Linux, Android and Microsoft Windows.
2. Application Software: Application Software is a computer program designed to carry out a specific task as per user & business need.
 - Examples: Social medias apps, Gaming apps, Word processing apps, Multimedia apps, Banking apps, Shopping apps, Booking apps etc

Nature of Software

3. Engineering and Scientific Software: This software is used to facilitate the engineering function and task take real time.

- It has very high accuracy, complex formula evolution & data analysis.
- Examples: Weather prediction apps, Stock Market apps, Stress Analysis, Body measurement apps

4. Embedded Software: Embedded software resides within the system or product and is used to implement and control feature and function for the end-user and for the system itself.

- Example: Switches, Routers, Digital camera, Washing machine functionalities, Traffic control

Nature of Software

5. Web Applications: It is a client-server computer program which the client runs on the web browser.

- Web apps can be little more than a set of linked hypertext files that present information using text and limited graphics.
- Examples: Online forms, Shopping carts, Gmail, Yahoo, Photo editing, File conversion etc.

6. Artificial Intelligence Software: It makes use of a nonnumerical algorithm to solve a complex problem.

- Application within this area includes robotics, expert system, pattern recognition, artificial neural network, theorem proving and game playing.
- Examples: Google Cloud, Azure studio, Tensor Flow, Salesforce etc.

Software engineering fundamentals

- Planned and managed development process
- Software must behave as expected, always available and work without failure (dependable).
- Efficient use of resources (performance).
- Manage stakeholder expectations (requirements)
- Manage software reuse

Changing Nature of Software

WebApps

- Semantic web technologies(Web3.0)
- Semantic databases; web linking; flexible data representation; and application programmer interfaces(API's)

Mobile Apps

- Can gain direct access to the hardware found on the device to provide local processing and storage capabilities

Cloud Computing

- Cloud computing provides distributed data storage and processing resources to networked computing devices

Changing Nature of Software

Product Line Software

- A software product line shares a set of assets that include requirements, architecture, design patterns, reusable components, test cases and other work products

Service orientated software

- Software system is composed of standalone web-services

Rich internet interfaces

- AJAX and HTML5

Software engineering ethics

Software engineers must behave in an honest and ethically

- Be respected as professionals
- Ethical behavior is more than simply upholding the law
- involves following a set of principles that are morally correct.
- Respect Confidentiality of employers and clients
- Not accept work outside your Competence
- Protect the intellectual property of employers and clients
- Not use technical skills to misuse other people's computers
- ACM/IEEE maintain code of ethics and professional practice

<https://www.youtube.com/watch?v=d91MTV9AINg>

Software Process Activities

- A software process is the set of activities and associated outcome that produce a software product.
- Software engineers mostly carry out these activities.
 1. **Software Specifications:** The functionality of the software and constraints on its operation must be defined
 2. **Software Development:** The software to meet the requirement must be produced.
 3. **Software Validation:** The software must be validated to ensure that it does what the customer wants.
 4. **Software Evolution:** The software must evolve to meet changing client needs.

SDLC Models

