

# GRADE 11 ICT – Final STUDY GUIDE

(Covers Python + Unit 2 Lesson 1)

## PART 1: PYTHON BASICS

### 1. Variables and Data Types

A variable is a container used to store information in a program.

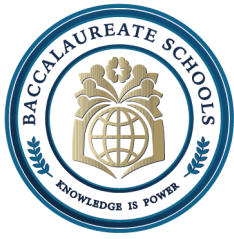
Examples:

```
name = "Ahmad"
```

```
age = 15
```

Rules for naming variables:

- Only letters, numbers, and underscore
- Cannot start with a number
- Cannot contain spaces
- Cannot use Python keywords (if, for, print, etc.)
- Use clear names (student\_name, total\_price)



Common data types:

- int: whole numbers (1, 20, 300)
- float: decimal numbers (2.5, 0.1, 89.99)
- str: text inside quotes ("hello")
- list: multiple values in brackets

Check type:

`type(age)`

## 2. Strings

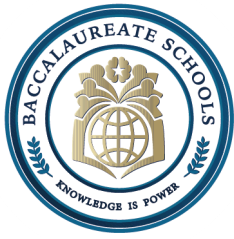
Strings are text written inside quotation marks.

Example:

`message = "Hello"`

Useful string methods:

- `upper()`: converts to uppercase
- `lower()`: converts to lowercase



- `title()`: capitalizes each word
- `strip()`: removes spaces from both sides
- `lstrip()`: removes left spaces
- `rstrip()`: removes right spaces

Example:

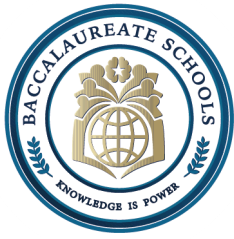
```
" hello ".strip()    # "hello"  
"world".upper()     # "WORLD"
```

String formatting (f-strings):

```
name = "Lana"  
print(f"My name is {name}")
```

Special characters:

- `\n` = new line
- `\t` = tab



### 3. Numbers and Operators

Number types:

- int
- float

Arithmetic operators:

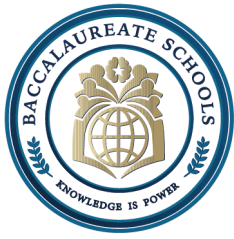
- addition
- subtraction
- multiplication
- / division
- % remainder
- \*\* exponent (power)

Examples:

$5 + 3$     # 8

$7 \% 2$     # 1

$2 ** 3$     # 8



## 4. Input from the User

The input() function allows the user to enter data.

Example:

```
name = input("Enter your name: ")
```

**Input is always a string. To convert:**

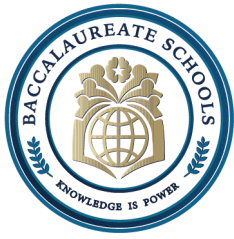
```
age = int(input("Enter age: "))
```

```
price = float(input("Enter price: "))
```

## 5. Conditionals (if statements)

Comparison operators:

- == equal
- != not equal
- greater
- < less
- = greater or equal
- <= less or equal



### Basic if:

```
if age >= 18:  
    print("Adult")
```

### If-else:

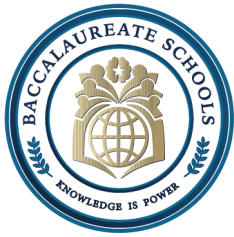
```
if grade >= 50:  
    print("Pass")  
else:  
    print("Fail")
```

### If-elif-else:

```
if score >= 90:  
    print("A")  
elif score >= 80:  
    print("B")  
else:  
    print("C or below")
```

### Logical operators:

- and
- or
- not



Example:

```
if color == "red" or color == "blue":  
    print("Primary color")
```

Check membership:

```
if "apple" in fruits:  
    print("Found")
```

## 6. Loops

### For loop

Used when repeating a known number of times.

Example:

```
for i in range(5):  
    print(i)
```

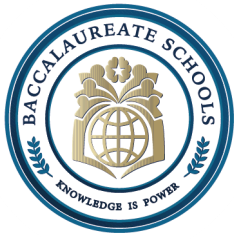
Range with start and end:

```
for i in range(1, 6):  
    print(i)
```

### While loop

Runs while a condition is true.

Example:



```
count = 1
while count <= 5:
    print(count)
    count += 1
```

### **Nested loops**

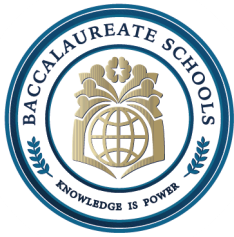
A loop inside a loop.

```
for i in range(3):
    for j in range(2):
        print(i, j)
```

## **7. Using Google Colab**

Google Colab allows:

- Running Python online
- No installation needed
- Automatic saving in Google Drive
- Suitable for school assignments



## Unit 2 – Lesson 1: AI & Data Science + Image Processing with Pillow

---

### PART 1 — AI IN SOFTWARE AND HARDWARE

#### 1. How AI Appears in Devices

AI is integrated into many devices you use daily (phones, speakers, home systems).

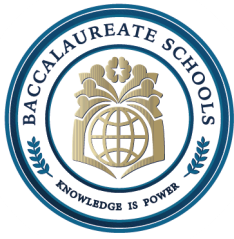
Devices use both:

- **Software components** (algorithms, apps, voice recognition)
- **Hardware components** (sensors, microphones, cameras)

These two work together to make devices smart.

Examples from the text:

- Smart speakers understand voice commands
- Phones detect faces
- Home systems automate lighting, security, and temperature



## 2. AI Applications and Their Evolution

### a. Smart Home Systems

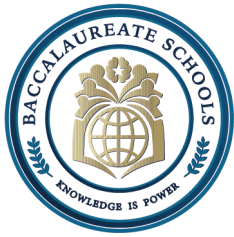
Smart homes include:

- Smart locks
- Smart thermostats
- Environmental sensors
- Automatic lighting
- Motion detectors
- Security cameras

These devices communicate with each other using AI and DS.

AI allows the system to:

- Detect unusual behavior
- Control lights and temperature
- Recognize faces to unlock the door (Face ID)
- Send security alerts
- Reduce energy use by learning habits



## **b. AI in Face Recognition**

Smartphones scan the user's face using AI

The system analyzes:

- Patterns
- Landmarks
- Depth
- Unique features

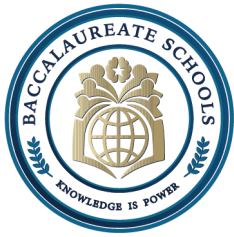
If the match is correct → the phone unlocks.

This increases:

- Privacy
- Security
- User convenience

## **c. AI in Voice Assistants**

- Natural language processing
- Machine learning models



They understand commands such as:

- Turning off lights
- Playing music
- Checking the weather
- Setting reminders

#### **d. AI in Factories and Industry**

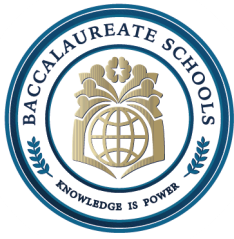
- Automated assembly lines
- Robots
- Quality control
- Predictive maintenance

AI helps reduce human errors and increases efficiency.

#### **e. AI in Healthcare**

- AI analyzes medical images like CT scans and MRIs
- Detects diseases earlier than humans
- Helps doctors diagnose accurately

This improves healthcare decision-making.



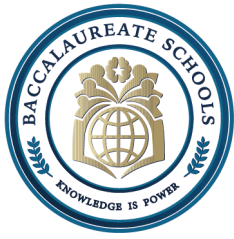
## PART 2 — AI VS NON-AI FEATURES

### Examples of AI-related smartphone features:

- Face recognition
- Intelligent photo enhancement
- Auto-categorizing images
- Predictive typing
- Battery optimization
- Voice assistant behavior

### Examples of non-AI features:

- Basic calling
- Simple texting
- Standard hardware functions
- No learning ability



- Using a smart lock using facial recognition → AI
- Detecting defects in a manufacturing plant → AI
- Searching with Google Lens → AI
- Using a calculator → not AI
- Sending a normal SMS → not AI

## PART 4 — BASIC COMPUTER VISION USING PILLOW (PIL)

### 1. Loading Images with Pillow

To use Pillow:

```
from PIL import Image
```

To open an image:

```
image = Image.open("image.jpg")
```

The image appears in the output.

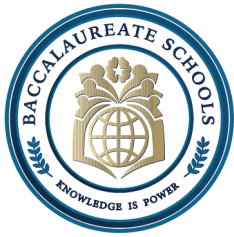
### 2. Exploring Image Properties

Using attributes:

```
print(image.size)    # width, height
```

```
print(image.format)  # JPEG, PNG...
```

```
print(image.mode)    # RGB, L, etc.
```



example outputs:

- Width: 1000
- Height: 667
- Format: JPEG
- Mode: RGB

### 3. Image Preprocessing

Preprocessing prepares images for analysis

#### a. Converting to Grayscale

```
gray = image.convert("L")
```

This removes color and keeps intensity only.

Uses:

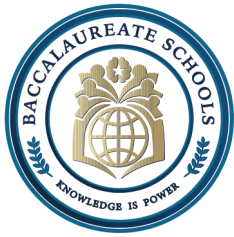
- Easier edge detection
- Reducing data size

#### b. Resizing Images

Important for ML models that require fixed dimensions.

Example from page 10:

```
resized = image.resize((300, 100))
```



### c. Rotating Images

```
rotated = image.rotate(45)
```

You can rotate to any angle.

### 4. Edge Detection Using Pillow

Edge detection identifies boundaries and outlines in an image.

It shows sharp changes in brightness.

To do this, import Pillow's built-in filter:

```
from PIL import ImageFilter
```

```
edges = image.filter(ImageFilter.FIND_EDGES)
```

### KEY TERMS (FOR EXAMS)

1. **AI (Artificial Intelligence):** Machines performing tasks that need human intelligence.
2. **Data Science:** Using data to extract meaningful patterns.
3. **Smart Home:** Home using sensors and AI to automate actions.
4. **Face Recognition:** AI technique to identify people from facial features.
5. **Pillow (PIL):** Python library for image processing.
6. **Grayscale:** Image with shades of gray only.
7. **Resizing:** Changing image dimensions.
8. **Edge Detection:** Identifying boundaries in an image.
9. **RGB:** Color mode (Red, Green, Blue).
10. **Format:** Type of image (JPEG, PNG...).