

Inception BD

111 no House, Domdom Sharok, Lahini Bot Tola, Kushtia-7001

Full Stack Data Science with Generative AI Bootcamp 1.0

Assignment 03

Last Submission Date: 23 November 2025, at 11:59 PM

Assignment Submission Instructions:-

Step 1: Create a new **repository** on your GitHub account.

Step 2: **Upload your code** (the `.ipynb` file) from your computer to that repository.

Step 3: **Copy the link to your `.ipynb` file.**

Step 4: **Send the link via email to:** learn.inceptionbd@gmail.com

Gmail Format:-

Compose a new email with the following details:

To: learn.inceptionbd@gmail.com

Subject: FSDS with Gen AI Assignment-3 Submission

Body:

Hello, please find my assignment submission at the following GitHub repository link.

Name: Md. Ridoy Hossain

Student ID: 2AE89DFB

Repository Link: <https://github.com/inceptionbd/Computer-Vision>

Best Regards

Md. Ridoy Hossain

1. Inventory Price Calculator

Real-world use: Used in e-commerce dashboards for price summaries.

Task:

Write a function `calculate_inventory_summary(prices)` that:

- receives a list of product prices
- returns total_cost and average_cost

Example:

Input: [120, 250, 399, 150]

Output: (919, 229.75)

2. Unique Visitor Tracker

Real-world use: Website analytics tools calculating daily active users.

Task:

Write a function `count_unique_visitors(visitor_list)` that:

- takes a list of visitor IDs
- returns the number of unique visitors

Example:

Input: [101, 205, 101, 310, 205, 550]

Output: 4 (The unique IDs are 101, 205, 310, 550)

3. Product Category Counter

Real-world use: Inventory/category-based analytics.

Task:

Write a function `count_categories(products)` that:

- takes a list of product names
- returns a dictionary counting each product type.

Example:

Input: ["Laptop", "Smartphone", "Monitor", "Laptop", "Smartphone", "Headphones", "Laptop"]

Output: {"Laptop": 3, "Smartphone": 2, "Monitor": 1, "Headphones": 1}

4. Temperature Analytics

Real-world use: Health tech temperature screening systems.

Task:

Write a function `analyze_temperatures(temp_list)` that:

- separates high ($> 38^{\circ}\text{C}$) and normal temps
- returns two lists: (`high_list`, `normal_list`)

Example:

Input: [36.5, 38.0, 39.1, 37.2, 40.5, 38.0, 37.9]

Output:

High List ($> 38^{\circ}\text{C}$): [39.1, 40.5]

Normal List ($< 38^{\circ}\text{C}$): [36.5, 38.0, 37.2, 38.0, 37.9]

Full Output Tuple: ([39.1, 40.5], [36.5, 38.0, 37.2, 38.0, 37.9])

5. Menu Price Lookup

Real-world use: Restaurant ordering apps.

Task:

Write a function `get_price(menu, item)` that:

- receives a dictionary menu and an item name
- returns the price if exists
- otherwise returns "Item not found"

Example 1: Item Found

- Input (Menu): {"Coffee": 3.50, "Tea": 2.75, "Muffin": 4.00, "Sandwich": 8.50}
- Input (Item): "Muffin"
- Output: 4.00

Example 2: Item Not Found

- Input (Menu): {"Coffee": 3.50, "Tea": 2.75, "Muffin": 4.00, "Sandwich": 8.50}
- Input (Item): "Donut"
- Output: "Item not found"

6. Student Grading System

Real-world use: School result automation.

Task:

Write a function `calculate_grade(score)` that:

- returns "A+", "A", "A-", "F" based on numeric marks

Write another function `grade_students(student_dict)` to:

- take a dictionary of student:score
- return student:grade dictionary

Example:

Input (Scores Dictionary): `{"Alice": 85, "Bob": 78, "Charlie": 32, "David": 69}`

Output (Grades Dictionary): `{"Alice": "A+", "Bob": "A", "Charlie": "F", "David": "A-"}`

7. Discount Calculator

Real-world use: Online store discount engine.

Task:

Write a function `apply_discounts(products)` where:

- `products` is a list of tuples: `(name, price, discount)`
- return a list of updated prices after discount

Example:

Input (List of Tuples): `[("Keyboard", 80.00, 20), ("Mouse", 25.00, 10), ("Monitor", 300.00, 5)]`

Output (List of Prices): `[64.00, 22.50, 285.00]`

8. Hashtag Generator

Real-world use: Social media automation tools.

Task:

Write a function `generate_hashtags(sentence)` that:

- converts every word into a hashtag
- returns a list of hashtags

Example:

Input: "AI Data Science"

Output: `["#ai", "#data", "#science"]`

9. User Authentication Mock

Real-world use: Login systems (basic level).

Task:

Write a function `login(username, password)` that:

- checks if username exists
- checks if password matches
- returns "Login Successful" or "Invalid Credentials"

Example Setup (Mock Database): Assume the following credential database is used internally:

```
CREDENTIALS = {"alex_s": "securepwd1", "maria_t": "password101", "john_d": "johndoe123"}
```

Example 1: Successful Login

- Input (Username): "maria_t"
- Input (Password): "password101"
- Output: "Login Successful"

Example 2: Failed Login (Wrong Password)

- Input (Username): "alex_s"
- Input (Password): "securepwd12"
- Output: "Invalid Credentials"

Example 3: Failed Login (Non-existent User)

- Input (Username): "peter_z"
- Input (Password): "any_password"
- Output: "Invalid Credentials"

10. Shopping Cart Bill Calculator

Real-world use: Billing systems in e-commerce apps.

Task:

Write a function `calculate_total(cart, prices)` that:

- takes a shopping cart list
- takes a dictionary of item:price
- returns the total bill amount

Example:

Input (Cart): ["Milk", "Bread", "Milk", "Cheese", "Bread"]

Input (Price): {"Milk": 3.00, "Bread": 2.50, "Eggs": 4.00, "Cheese": 5.00}

Output (Total Bill Amount): 16.00