Problem 1.

You are designing an **e-commerce shopping cart** system that allows users to add multiple products. Each product is represented as [product_name, quantity, price] in an array.

Task:

- Implement a method that calculates the **total cart value**.
- If any product has a quantity greater than **5**, apply a **10% discount** to that product.
- Return the **final total amount**.

Sample Input:

cart = [["Laptop", 1, 800], ["Headphones", 6, 50], ["Mouse", 3, 25]]

Sample Output:

Final total: $1105.0

Problem 2:

Write a Ruby program that uses arrays to manage a simple student gradebook. The program should allow users to perform the following operations:

Requirements:

- Initially, store data for three students, each having:
    - A name (string)
    - A student ID (string)
    - An array of grades (integers)
- Implement functionality to:
    - Add a new student with their grades.
    - Update grades for an existing student.
    - Remove a student from the gradebook using their student ID.
    - Calculate and display the average grade for a specific student.
    - Display a subset of students using array slicing (by specifying start and end indices).
    - Display the full gradebook with student details and their average grades.

Initial Sample Data (Stored in an Array):

students = [["Alice", "S001", [85, 90, 78]], ["Bob", "S002", [75, 80, 88]], ["Charlie", "S003", [90, 92, 95]]]

Expected Output:

Enter the new student's name: David

Enter the student's ID: S004

Enter the student's grades: 88 92 85

New student added!

Enter the student ID to update grades: S002

Enter the new grades: 80 85 90

Student grades updated!

Enter the student ID to remove: S001

Removed student with ID: S001

Student removed successfully!

Enter the student ID to calculate average grade: S003

Average grade for Charlie: 92.33

Enter the starting and ending index of students you want to view: 0 1

Bob (ID: S002) - Average Grade: 85.00

Charlie (ID: S003) - Average Grade: 92.33

Gradebook:

Name	Student ID	Grades	Average Grade
Bob	S002	80, 85, 90	85.00
Charlie	S003	90, 92, 95	92.33
David	S004	88, 92, 85	88.33

Problem 3:

You are tasked with creating a **Ruby program** to manage a **library system** using **hashes**. The program should allow users to store, modify, and retrieve book information efficiently.

Requirements:

1. **Use a hash** to store book data, where:
    - The **ISBN number** is the **key** (string).
    - The **value** is another hash containing:
        - **Title** (string)
        - **Author** (string)
        - **Copies Available** (integer)
2. The program should provide options to:
    - **Add a new book** to the library.
    - **Update the number of copies** available for an existing book.
    - **Remove a book** from the library using its ISBN.
    - **Search for a book** by ISBN and display its details.
    - **List all books** in the library with their details.

Initial Sample Data (Stored in a Hash):

```
library = {
  "978-0143127741" => { title: "The Alchemist", author: "Paulo Coelho", copies: 5 },
  "978-0062315007" => { title: "Sapiens", author: "Yuval Noah Harari", copies: 3 },
  "978-0451524935" => { title: "1984", author: "George Orwell", copies: 4 }
}
```

Expected Sample Interaction:

```
Enter the new book's title: Atomic Habits
Enter the author's name: James Clear
Enter the ISBN: 978-0735211292
Enter the number of copies: 7
New book added!

Enter the ISBN to update copies: 978-0062315007
Enter the new number of copies: 5
Book copies updated!

Enter the ISBN to remove: 978-0451524935
Removed book with ISBN: 978-0451524935
Book removed successfully!

Enter the ISBN to search: 978-0143127741
```

Book Details:
Title: The Alchemist
Author: Paulo Coelho
Copies Available: 5

Library Catalog:

ISBN	Title	Author	Copies
978-0143127741	The Alchemist	Paulo Coelho	5
978-0062315007	Sapiens	Yuval Noah Harari	5
978-0735211292	Atomic Habits	James Clear	7