

1. Store and display student information (name, age, grade).

Coding Python

Auto saved at 11:57:25

```
1 def student_info():
2     student = ("Ammara", 18, "A")
3     print("Name:", student[0])
4     print("Age:", student[1])
5     print("Grade:", student[2])
6
7 student_info()
```

Compile Result

Name= Ammara
Age= 18
Grade= A

[Process completed - press Enter]

2. List Prices of Grocery Items and Total Them

Coding Python

Auto saved at 12:01:49

```
1 def grocery_prices():
2     prices = (25.5, 40.0, 60.25, 10.75)
3     print("Prices:", prices)
4     print("Total Price:", sum(prices))
5 grocery_prices()
```

Compile Result

```
Prices: (25.5, 40.0, 60.25, 10.75)
Total Price: 136.5
```

```
[Process completed - press Enter]
```

3. Pair Items with Prices

Coding Python RUN MENU
Auto saved at 12:05:05

```
1 def item_price_pairs():
2     items = (("Milk", 25), ("Eggs", 50), ("Bread", 20))
3     for item in items:
4         print(item[0], ":", item[1], "Rs")
5 item_price_pairs()
```

Compile Result

Milk : 25 Rs
Eggs : 50 Rs
Bread : 20 Rs

[Process completed - press Enter]

4. Store and Display Train Schedule

Coding Python RUN MENU
Auto saved at 12:08:36

```
1 def train_schedule():
2     trains = (("Ammara", "10:00"), ("sadiya", "12:30"), ("sara", "17:00"))
3     for train in trains:
4         print("Train:", train[0], "| Time:", train[1])
5 train_schedule()
```

Compile Result

```
Train= Ammara | Time= 10:00
Train= sadiya | Time= 12:30
Train= sara | Time= 17:00
```

```
[Process completed - press Enter]
```

5. Sort Employee Records by Salary

Coding Python RUN MENU
Auto saved at 12:11:55

```
1 def sort_employees():
2     employees = (("Ammara", 40000), ("sadiya", 55000), ("sara", 30000))
3     sorted_emp = sorted(employees, key=lambda x: x[1], reverse=True)
4     for emp in sorted_emp:
5         print(emp[0], "-", emp[1])
6 sort_employees()
```

Compile Result

```
sadiya - 55000
Ammara - 40000
sara - 30000
```

[Process completed - press Enter]

6. Count Students Scoring Above 75

Coding Python

Auto saved at 12:14:39

```
1 def count_high_scores():
2     marks = (67, 88, 92, 74, 76, 55)
3     count = sum(1 for mark in marks if mark > 75)
4     print("Students scoring above 75:", count)
5 count_high_scores()
```

Compile Result

Students scoring above 75: 3

[Process completed - press Enter]

7. Find Maximum Stock Price

Coding Python

Auto saved at 12:17:16

```
1 def max_stock_price():
2     prices = (154.5, 160.2, 149.8, 170.1)
3     print("Maximum stock price:", max(prices))
4 max_stock_price()
```

Compile Result

Maximum stock price= 170.1

[Process completed - press Enter]

8. Average Temperature of the Day

Coding Python RUN

Auto saved at 12:20:11

```
1 def average_temperature():
2     temps = (29.5, 30.0, 32.2, 31.5, 28.9)
3     avg = sum(temps) / len(temps)
4     print("Average temperature:", round(avg, 2), "°C")
5 average_temperature()
```

Compile Result

Average temperature= 30.42 °C

[Process completed - press Enter]

9. Schedule Appointments

Coding Python

RUN

Auto saved at 12:29:57

```
1 def appointments():
2     schedule = (("Doctor", "10:00 AM"), ("Meeting", "2:00 PM"))
3     for task in schedule:
4         print("Appointment:", task[0], "| Time:", task[1])
5 appointments()
```

Compile Result

```
Appointment= Doctor | Time= 10:00 AM
Appointment= Meeting | Time= 2:00 PM
```

```
[Process completed - press Enter]
```

10. Store Contact Information

Coding Python

Auto saved at 12:33:31

RUN

MENU

```
1 def contacts():
2     contact_list = (("Ammara", "9876543210"), ("Ammo", "9123456780"))
3     for contact in contact_list:
4         print("Name:", contact[0], "| Phone:", contact[1])
5 contacts()
```

Compile Result

Name= Ammara | Phone= 9876543210
Name= Ammo | Phone= 9123456780

[Process completed - press Enter]

11. Display Exam Schedule

Coding Python

Auto saved at 12:45:03

RUN

MENU

```
f exam_schedule():
    exams = (("Math", "9:00 AM"), ("Science", "11:30 AM"), ("English", "2:00 PM"))
    for subject in exams:
        print("Subject:", subject[0], "| Time:", subject[1])
am_schedule()
```

Compile Result

```
Subject= Math | Time= 9:00 AM
Subject= Science | Time= 11:30 AM
Subject= English | Time= 2:00 PM
```

```
[Process completed - press Enter]
```