

TASK 2

Build a Personal Information Manager that stores and displays your name, age, city, and hobbies with formatted output.

CODE:

```
results = []

def calculate_grade(marks):

    if marks >= 90:
        return "A+", "Outstanding performance!"

    elif marks >= 80:
        return "A", "Great job! Keep it up."

    elif marks >= 70:
        return "B", "Good work. You can do even better."

    elif marks >= 60:
        return "C", "Fair effort. Try to improve."

    elif marks >= 50:
        return "D", "Needs improvement. Keep practicing."

    else:
        return "F", "Fail. Don't give up—study harder!"

while True:

    name = input("Enter student name: ")

    marks = float(input("Enter marks (0-100): "))

    grade, comment = calculate_grade(marks)

    results.append({
        "Name": name,
        "Marks": marks,
        "Grade": grade,
        "Comment": comment
    })

    print(f"\n {name}'s Result:")
    print(f"Marks: {marks}")
```

```

print(f"Grade: {grade}")

print(f"Comment: {comment}\n")

choice = input("Add another student? (yes/no): ").lower()

if choice != "yes":
    break

print("\n all Student Results:")

for student in results:
    print(student)

```

OUTPUT:

The screenshot shows the Python Tutor interface running on Python 3.11. The code in the editor calculates grades based on marks and prints student results. The console output shows the grade for Vasanth (B) and asks if another student is added. The Global frame and Objects panel show the current state of variables.

Python 3.11
known_imitations

```

1 results = []
2 def calculate_grade(marks):
3     if marks >= 90:
4         return "A+", "Outstanding performance!"
5     elif marks >= 80:
6         return "A", "Great job! Keep it up."
7     elif marks >= 70:
8         return "B", "Good work. You can do even better."
9     elif marks >= 60:
10        return "C", "Fair effort. Try to improve."
11    elif marks >= 50:
12        return "D", "Needs improvement. Keep practicing."
13    else:
14        return "F", "Fail. Don't give up-study harder!"
15
16 while True:
17     name = input("Enter student name: ")
18     marks = float(input("Enter marks (0-100): "))
19
20     grade, comment = calculate_grade(marks)
21     results.append((name, marks, grade, comment))

```

Print output (drag lower right corner to resize)

Enter student name: Vasanth
Enter marks (0-100): 70

Vasanth's Result:
Marks: 70.0
Grade: B
Comment: Good work. You can do even better.

Add another student? (yes/no): Dinesh

all Student Results:
{'Name': 'Vasanth', 'Marks': 70.0, 'Grade': 'B', 'Comment': 'Good work. You can do even better.'}

Global frame

results	
calculate_grade	
name	"Vasanth"
marks	70.0
grade	"B"
comment	"Good work. You can do even better."
choice	"dinesh"
student	

Objects

list	0
function	calculate_grade(marks)
dict	"Name" "Vasanth" "Marks" 70.0 "Grade" "B" "Comment" "Good work. You can do even better."

<<First <Prev Next >> Last >>

Done running (29 steps)

28°C Mostly cloudy Search 11:26 ENG IN 22-11-2025