Python Basics Workshop

Exercise Set 1: Understanding Variables

Task 1: Variable Creation & Types

```
customer_num = 789
customer_full_name = "Taylor Swift"
premium_member = True
discount_percent = 15.5

# Output with type checking
print(customer_num, type(customer_num))
print(customer_full_name, type(customer_full_name))
print(premium_member, type(premium_member))
print(discount_percent, type(discount_percent))
```

Result shows: integer, string, boolean, and float types

Task 2: Type Conversion Examples

```
# Example A
num = 50
text_num = "50"
print(f"{num} is {type(num)}")
print(f"{text_num} is {type(text_num)}")

# Example B
value = 50
converted = str(50)
print(f"{value} is {type(value)}")
print(f"{converted} is {type(converted)}")
```

Task 3: Type Prediction Challenge Predict these types:

- 87 (integer)

- 5.25 (float)
- "coding" (string)
- True (boolean)

Task 4: Descriptive Variables Practice

```
pupil_name = 'emma'
final_exam_score = 88
passed_course = True
```

Task 5: Rules for Variable Names

Rule	Valid Example	Invalid Example	Error Type
Start with letter/underscore	`total_sum = 500`	`5total = 500`	SyntaxError
Allow letters/numbers/underscores	`user_id_2023 = "abc123"`	*n/a*	*n/a*
No special characters	`item_count = 42`	`item-count = 42`	SyntaxError
No reserved keywords	`my_class = "Physics"`	`class = "Physics"`	SyntaxError
Case sensitivity matters	`firstName = "John"`	*both valid but different*	*n/a*
Define before using	`age = 25; print(age)`	`print(height)` (undefined)	NameError

Task 6: Narrative with Variables

```
# Character details
protagonist = "Harry Potter"
antagonist = "dark wizard"
location = "Hogwarts"
book_title = "Sorcerer's Stone"

# Numeric elements
spells_cast = 12
magical_items = 3
adventure_year = 1991
```

```
# Creating the story
print(protagonist, "confronted the", antagonist, "at", location,
"using", spells_cast, "different spells.")
print("He discovered", magical_items, "magical artifacts in the",
book_title, "adventure that took place in", adventure_year, ".")
```

Task 7: Variable Value Exchange

```
# Using temporary storage
first_value = 42
second_value = 18

temp = second_value
second_value = first_value
first_value = temp
print(f"After swap: {first_value}, {second_value}")

# Python's direct exchange method
first_value = 42
second_value = 18

first_value, second_value = second_value, first_value
print(f"After pythonic swap: {first_value}, {second_value}")
```