Answers to Practice Problems – Lesson 4

Solutions

1. Create a list of 5 numbers and print the largest using a loop.

```
nums = [12, 7, 25, 9, 18]

largest = -float('inf')
for n in nums:
    if n > largest:
        largest = n

print("Largest number is:", largest)
```

Output

```
Largest number is: 25
```

2. Make a list of 3 prices, then use a loop to print only those under budget=10.

```
prices = [12, 8, 5]
budget = 10

for p in prices:
    if p < budget:
        print("Affordable:", p)
```

Output

```
Affordable: 8
Affordable: 5
```

3. Store 3 routes as tuples (time, fun) and print the one with the best fun score.

```
routes = [(12, 8), (15, 10), (9, 6)]
best_fun = -float('inf')
best_route = None

for t, f in routes:
    if f > best_fun:
```

```
        best_fun = f
        best_route = (t, f)

print("Best route by fun:", best_route)
```

Output

```
Best route by fun: (15, 10)
```

4. Make a dictionary of 3 snacks with their prices and print the cheapest one.

```
snacks = {"apple": 2, "banana": 1, "chips": 3}
cheapest_item = None
cheapest_price = float('inf')

for snack, price in snacks.items():
    if price < cheapest_price:
        cheapest_price = price
        cheapest_item = snack

print("Cheapest snack:", cheapest_item, "at", cheapest_price)
```

Output

```
Cheapest snack: banana at 1
```

5. Challenge: From a dictionary of (price, fun) pairs, pick the most fun item within budget=12.

```
items = {
    "board game": (10, 7),
    "book": (8, 6),
    "video game": (15, 10)
}
budget = 12
best_item = None
best_fun = -float('inf')

for name, (price, fun) in items.items():
    if price <= budget and fun > best_fun:
        best_fun = fun
        best_item = name

print("Best item within budget:", best_item, "with fun =", best_fun)
```

Output

```
Best item within budget: board game with fun = 7
```