```python
# 1. Find second smallest number in list
def second_smallest(lst):
    unique_nums = sorted(set(lst))
    return unique_nums[1] if len(unique_nums) > 1 else None

print(second_smallest([10, 5, 8, 20, 5, 12]))
```

```
8
```

```python
# 2. Capitalize first letter of each word
def capitalize_sentence(s):
    return s.title()

print(capitalize_sentence("hello world from python"))
```

```
Hello World From Python
```

```python
# 3. Flatten a nested dictionary
def flatten_dict(nested):
    flat = {}
    for k, v in nested.items():
        if isinstance(v, dict):
            flat.update(flatten_dict(v))
        else:
            flat[k] = v
    return flat

print(flatten_dict({'a': 1, 'b': {'c': 2, 'd': 3}}))
```

```
{'a': 1, 'c': 2, 'd': 3}
```

```python
# 4. Print first 10 prime numbers
def first_10_primes():
    primes = []
    num = 2
    while len(primes) < 10:
        for i in range(2, int(num**0.5) + 1):
            if num % i == 0:
                break
        else:
            primes.append(num)
        num += 1
    print(primes)

first_10_primes()
```

```
[2, 3, 5, 7, 11, 13, 17, 19, 23, 29]
```

```python
# 5. Count words starting with prefix
def count_prefix(words, pre):
    return sum(1 for w in words if w.startswith(pre))

print(count_prefix(['apple', 'ape', 'apricot', 'banana'], 'ap'))
```

```
3
```

```
# 6. Generate Fibonacci up to N terms
def fibonacci(n):
    a, b = 0, 1
    res = []
    for _ in range(n):
        res.append(a)
        a, b = b, a + b
    print(*res)

fibonacci(7)
```

```
0 1 1 2 3 5 8
```

```
# 7. Return index of first occurrence
def find_index(lst, n):
    return lst.index(n) if n in lst else -1

print(find_index([2, 4, 6, 8, 6], 6))
```

```
2
```

```
# 8. Reverse words in a sentence
def reverse_words(s):
    return " ".join(s.split()[::-1])

print(reverse_words("Hello World"))
```

```
World Hello
```

```
# 9. Count occurrence of each character
def char_count(s):
    return {c: s.count(c) for c in s}

print(char_count("hello"))
```

```
{'h': 1, 'e': 1, 'l': 2, 'o': 1}
```

```
# 10. Find all sublists of a list
def get_sublists(lst):
    res = [[]]
    for i in range(len(lst)):
        for j in range(i + 1, len(lst) + 1):
            res.append(lst[i:j])
    return res

print(get_sublists([1, 2, 3]))
```

```
[[], [1], [1, 2], [1, 2, 3], [2], [2, 3], [3]]
```