```python
def demonstrate_lists():
    empty_list = []
    print("Empty list:", empty_list)

    one_element_list = [42]
    print("List with one element:", one_element_list)

    identical_elements_list = [7] * 5
    print("List with all identical elements:", identical_elements_list)

    negative_numbers_list = [-1, -2, -3, -4, -5]
    print("List with negative numbers:", negative_numbers_list)

demonstrate_lists()

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS C:\Users\gudis\OneDrive\Desktop> & C:/Users/gudis/AppData/Local/Programs/Python/Python312/py
.py"
Empty list: []
List with one element: [42]
List with all identical elements: [7, 7, 7, 7, 7]
List with negative numbers: [-1, -2, -3, -4, -5]
PS C:\Users\gudis\OneDrive\Desktop>
```

```python
def selection_sort(arr):
    n = len(arr)
    for i in range(n):
        min_index = i
        for j in range(i + 1, n):
            if arr[j] < arr[min_index]:
                min_index = j
        arr[i], arr[min_index] = arr[min_index], arr[i]
    return arr

array = [64, 25, 12, 22, 11]
sorted_array = selection_sort(array)
print(sorted_array)
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS C:\Users\gudis\OneDrive\Desktop> & C:/Users/gudis/AppData/Local/Programs/P
[11, 12, 22, 25, 64]
PS C:\Users\gudis\OneDrive\Desktop>
```

```python
def bubble_sort(arr):
    n = len(arr)
    for i in range(n):
        swapped = False
        for j in range(0, n - i - 1):
            if arr[j] > arr[j + 1]:
                arr[j], arr[j + 1] = arr[j + 1], arr[j]
                swapped = True
        if not swapped:
            break
    return arr

array = [64, 34, 25, 12, 22, 11, 90]
sorted_array = bubble_sort(array)
print(sorted_array)
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS C:\Users\gudis\OneDrive\Desktop> & C:/Users/gudis/AppData/Local/Progr
[11, 12, 22, 25, 34, 64, 90]
PS C:\Users\gudis\OneDrive\Desktop>
```

```python
def insertion_sort(arr):
    n = len(arr)
    for i in range(1, n):
        key = arr[i]
        j = i - 1

        while j >= 0 and arr[j] > key:
            arr[j + 1] = arr[j]
            j -= 1

        arr[j + 1] = key

    return arr

array = [64, 34, 25, 34, 12, 22, 11, 34]
sorted_array = insertion_sort(array)
print(sorted_array)
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS C:\Users\gudis\OneDrive\Desktop> & C:/Users/gudis/AppData/Local/Programs/P
[11, 12, 22, 25, 34, 34, 34, 64]
PS C:\Users\gudis\OneDrive\Desktop>
```

```python
def find_kth_missing(arr, k):
    missing_count = 0
    current = 1
    index = 0
    n = len(arr)

    while missing_count < k:
        if index < n and arr[index] == current:
            index += 1
        else:
            missing_count += 1
            if missing_count == k:
                return current
        current += 1

    return -1

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS C:\Users\gudis\OneDrive\Desktop> & C:/Users/gudis/AppData/Local/Programs/P
9
PS C:\Users\gudis\OneDrive\Desktop>
```

```python
def find_peak(nums):
    left, right = 0, len(nums) - 1

    while left < right:
        mid = left + (right - left) // 2

        if nums[mid] < nums[mid + 1]:
            left = mid + 1
        else:
            right = mid

    return left

nums = [1, 2, 3, 1]
peak_index = find_peak(nums)
print(peak_index)
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS C:\Users\gudis\OneDrive\Desktop> & C:/Users/gudis/AppData/Loc
2
PS C:\Users\gudis\OneDrive\Desktop>
```

```python
def str_str(haystack, needle):

    if not needle:
        return 0

    return haystack.find(needle)


haystack = "hello"
needle = "ll"
index = str_str(haystack, needle)
print(index)

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS C:\Users\gudis\OneDrive\Desktop> & C:/Users/gudis/AppData/Loc
2
PS C:\Users\gudis\OneDrive\Desktop>
```

```python
def find_substrings(words):
    substrings = set()

    for i in range(len(words)):
        for j in range(len(words)):
            if i != j and words[i] in words[j]:
                substrings.add(words[i])

    return list(substrings)


words = ["mass", "as", "hero", "superhero", "her", "rat"]
result = find_substrings(words)
print(result)
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS C:\Users\gudis\OneDrive\Desktop> & C:/Users/gudis/AppData/Local/Programs/Pyth
['hero', 'her', 'as']
PS C:\Users\gudis\OneDrive\Desktop>
```