Q1. What are keywords in Python? Using the keyword library, print all the Python keywords.

Keywords in Python are reserved words that have a specific meaning and purpose in the language. They cannot be used as variable names, function names, or identifiers.

Here's the code to print all Python keywords using the `keyword` library:

```
import keyword
print(keyword.kwlist)
```

Q2. What are the rules to create variables in Python?

1. Variable names must start with a letter or an underscore (_).
2. Variable names can contain letters, digits, and underscores but cannot start with a digit.
3. Python keywords cannot be used as variable names.
4. Variable names are case-sensitive (var and VAR are different).
5. Avoid using special characters or spaces in variable names.

Q3. What are the standards and conventions for variable nomenclature in Python?

Python follows the PEP 8 guidelines for naming variables:

1. Use lowercase letters separated by underscores (snake_case).
2. Avoid single-character names unless used as iterators (e.g., i, j).
3. Use descriptive names that convey the variable's purpose.
4. Constants should be written in all caps (e.g., MAX_VALUE).
5. Private variables can be prefixed with an underscore (e.g., _private_var).

Q4. What will happen if a keyword is used as a variable name?

Using a keyword as a variable name will result in a SyntaxError.

Keywords are reserved for Python's internal syntax and cannot be redefined.

Q5. For what purpose is the `def` keyword used?

The `def` keyword is used to define a function in Python. Example:

```
def greet():
    print("Hello, world!")
```

Q6. What is the operation of this special character `\`?

The backslash (\) is used as an escape character in Python. It allows you to include special characters in strings. Examples:

-

for newline

- for tab

- \ to include a literal backslash

Q7. Examples of specific conditions:

1. Homogeneous list: A list with elements of the same data type.

   Example: [1, 2, 3, 4]

2. Heterogeneous set: A set with elements of different data types.

Example: {1, "apple", 3.14}

3. Homogeneous tuple: A tuple with elements of the same data type.

Example: ("apple", "banana", "cherry")

Q8. Explain mutable and immutable data types.

1. Mutable data types: Their content can be changed after creation.

Examples: List, Set, Dictionary

```
my_list = [1, 2, 3]
my_list[0] = 10  # List content changed
```

2. Immutable data types: Their content cannot be changed after creation.

Examples: Tuple, String, Int

```
my_tuple = (1, 2, 3)
# my_tuple[0] = 10  # This will raise an error
```

Q9. Write a code to create the given structure using only a for loop:

```
*
***
*****
*******
*********
```

```
rows = 5
for i in range(1, rows + 1):
    print("*" * (2 * i - 1))
```

Q10. Write a code to create the given structure using a while loop:

|||||||||

|||||||

|||||

|||

|

```
rows = 5
while rows > 0:
    print("|" * (2 * rows - 1))
    rows -= 1
```