

Q1. What are the two latest user-defined exception constraints in Python 3.X? ¶

In []:

```
1 Inheritance from BaseException
2 Exception chaining with __cause__ and __context__
```

Q2. How are class-based exceptions that have been raised matched to handlers?

In []:

```
1 It depends on Exception hierarchy and the order in which the handlers are defined.
2
3 Python searches for the nearest enclosing try-except block that can handle the speci
```

Q3. Describe two methods for attaching context information to exception artefacts.

In []:

```
1 Exception arguments :e.arg
2 exception attributes: e.context
```

Q4. Describe two methods for specifying the text of an exception object's error message.

```
1 For normal exception str(e)
2 Custom exceptions e.message
3
```

Q5. Why do you no longer use string-based exceptions?

In []:

```
1 Using better Exception handling and the use of good Exception Attributes and Method
2 string-based exceptions
```

