

Quiz 15 – Exceptions

1. Which keyword do you use to mark a section of code where something may go wrong?
(This was discussed in lecture “The begin and rescue Keywords”)

- a. unstable
- b. handle
- c. begin
- d. catch

Correct Answer: Option c - begin

Explanation:

Option a – Incorrect answer. Please try again.

Option b – Incorrect answer. Please try again.

Option c – Good job!

Option d – Incorrect answer. Please try again.

2. Which keyword do you use to handle an exception if one occurs?
(This was discussed in lecture “The begin and rescue Keywords”)

- a. deal
- b. rescue
- c. catch
- d. handle

Correct Answer: Option b - rescue

Explanation:

Option a – Incorrect answer. Please try again.

Option b – Good job!

Option c – Incorrect answer. Please try again.

Option d – Incorrect answer. Please try again.

3. What does the retry keyword do?
(This was discussed in lecture “The retry Keyword”)

- a. Exits the Ruby program.
- b. Skips to the next section of code.
- c. Reruns a section of a code.
- d. Raises another exception.

Correct Answer: Option c – Reruns a section of a code.

Explanation:

Option a – Incorrect answer. Please try again.

Option b – Incorrect answer. Please try again.

Option c – Good job!

Option d – Incorrect answer. Please try again.

4. Which block of code is guaranteed to run regardless of whether an exception was raised or not?

(This was discussed in lecture “The ensure Keyword”)

- a. begin
- b. retry
- c. rescue

d. ensure

Correct Answer: Option d - ensure

Explanation:

Option a – Incorrect answer. Please try again.

Option b – Incorrect answer. Please try again.

Option c – Incorrect answer. Please try again.

Option d – Good job!

5. Which keyword intentionally raises an exception in Ruby?
(This was discussed in lecture “The raise Keyword”)

- a. raise
- b. warn
- c. alert
- d. error

Correct Answer: Option a - raise

Explanation:

Option a – Good job!

Option b – Incorrect answer. Please try again.

Option c – Incorrect answer. Please try again.

Option d – Incorrect answer. Please try again.

6. If you want to define a custom error class in Ruby, which class should you inherit from?
(This was discussed in lecture “Custom Exceptions”)

- a. `Object`
- b. `StandardError`
- c. `BasicObject`
- d. `Exception`

Correct Answer: Option b - `StandardError`

Explanation

Option a – Incorrect answer. Please try again.

Option b – Good job!

Option c – Incorrect answer. Please try again.

Option d – Incorrect answer. Please try again.

7. In a rescue block, what syntax do you use to capture the exception object?
(This was discussed in lecture “Catching Error Object”)

- a. `rescue => e`
- b. `rescue :error`
- c. `rescue catch e`
- d. `handle exception as e`

Correct Answer: Option `rescue => e`

Explanation:

Option a – Good job!

Option b – Incorrect answer. Please try again.

Option c – Incorrect answer. Please try again.

Option d – Incorrect answer. Please try again.

8. Which of the following is NOT a recommended practice in Ruby exception handling?
- a. Rescue only the exceptions you expect.
 - b. Use ensure to close files or release resources.
 - c. Rescue all exceptions blindly without understanding them.
 - d. Define custom exceptions for specific error scenarios in your application.

Correct Answer: Option c - Rescue all exceptions blindly without understanding them.

Explanation:

Option a – Incorrect answer. Please try again.

Option b – Incorrect answer. Please try again.

Option c – Good job!

Option d – Incorrect answer. Please try again.