

Abschluss-Quiz

Das Quiz soll Ihnen einen ersten Hinweis auf Ihren Lernfortschritt geben. Nach unserer Einschätzung sollten Sie diese Fragen alle beantworten können, wenn Sie den Stoff der Woche durchgearbeitet und verstanden haben. Natürlich gibt es noch sehr viel mehr mögliche Fragen, dazu wollen wir auf die Literatur und das Internet verweisen. Geben Sie gerne einmal „python quizzes“ bei Google ein.

1. Wie erstellt man in Python einen Socket für eine TCP-Verbindung?

- a) `socket.socket(socket.AF_INET, socket.SOCK_STREAM)`
- b) `socket.tcp()`
- c) `tcp.socket()`
- d) `socket.create_tcp()`

2. Welche Methode wird in Python verwendet, um auf einem Socket zu lauschen (listen) und Verbindungen zu akzeptieren?

- a) `socket.accept()`
- b) `socket.listen()`
- c) `socket.bind()`
- d) `socket.connect()`

3. Welche Python-Bibliothek wird typischerweise für die Implementierung von HTTP-Anfragen verwendet?

- a) `requests`
- b) `http`
- c) `socket`
- d) `urllib`

4. In der Netzwerkprogrammierung mit Python, wofür wird `socket.bind()` verwendet?

- a) Zum Senden von Daten über einen Socket.
- b) Zum Verbinden eines Sockets mit einer Netzwerkadresse.
- c) Zum Anhören eingehender Anfragen auf einem bestimmten Port.
- d) Zum Akzeptieren einer Verbindung.

5. Was ist der Zweck der `send()` und `recv()` Methoden in der `socket`-Bibliothek in Python?

- a) `send()` zum Senden und `recv()` zum Empfangen von Daten über einen Socket.
- b) `send()` zum Verbinden und `recv()` zum Trennen von Sockets.
- c) `send()` zum Abfragen und `recv()` zum Empfangen von HTTP-Anfragen.

d) send() und recv() werden in Python nicht verwendet.

6. Wie erzeugt man in Python einen UDP-Socket?

a) socket.socket(socket.AF_INET, socket.SOCK_DGRAM)

b) socket.udp()

c) udp.socket()

d) socket.create_udp()

7. Was ist eine gängige Anwendung für das select Modul in der Python-Netzwerkprogrammierung?

a) Zum Parsen von HTML.

b) Für die gleichzeitige Überwachung mehrerer Sockets.

c) Zum Verschlüsseln von Daten.

d) Zur Verbesserung der Datenübertragungsgeschwindigkeit.

8. Wie wird in Python eine Verbindung von einem Client zu einem Server-Socket hergestellt?

a) socket.connect()

b) socket.bind()

c) socket.listen()

d) socket.accept()

9. Welche Aussage über das socket-Modul in Python ist korrekt?

a) Es unterstützt nur die TCP-Netzwerkprotokolle.

b) Es wird hauptsächlich für die Entwicklung von GUI-Anwendungen verwendet.

c) Es bietet Low-Level-Netzwerkfunktionen.

d) Es ist ein High-Level-HTTP-Client.

10. Was ist der Hauptunterschied zwischen socket.send() und socket.sendall() in Python?

a) send() sendet einmal, sendall() wiederholt das Senden bis alle Daten übertragen sind.

b) send() ist für UDP, sendall() für TCP.

c) Es gibt keinen Unterschied, beide Funktionen sind identisch.

d) send() benötigt eine Netzwerkadresse, sendall() nicht.