

# Python Exam Review

## Getting Started

```
print("Hello, Python!")
```

## Variables & Data Types

```
x = 5          # int
y = "Hello"    # str
z = 3.14       # float
is_active = True # bool
print(type(x)) # <class 'int'>
```

## Lists

```
fruits = ["apple", "banana", "cherry"]
print(fruits[1])
fruits.append("orange")
fruits[0] = "kiwi"
fruits.remove("banana")
```

## Looping Lists

```
for fruit in fruits:
    print(fruit)
```

## List Comprehension

```
squares = [x**2 for x in range(5)]
```

## Tuples

```
mytuple = ("apple", "banana", "cherry")
print(mytuple[0])
```

## Sets

```
myset = {"apple", "banana", "cherry"}
myset.add("orange")
myset.remove("banana")
```

## Dictionaries

```
person = {"name": "Alice", "age": 25}
print(person["name"])
person["age"] = 30
person["job"] = "engineer"
```

# Python Exam Review

## Conditions

```
x = 10
if x > 5:
    print("Greater")
elif x == 5:
    print("Equal")
else:
    print("Smaller")
```

## For Loop

```
for i in range(5):
    print(i)
```

## While Loop

```
i = 0
while i < 5:
    print(i)
    i += 1
```

## range()

```
for x in range(1, 10, 2):
    print(x)
```

## Functions

```
def greet(name):
    return f"Hello, {name}!"

print(greet("Alice"))
```

## Lambda Function

```
square = lambda x: x * x
print(square(4))
```

## User Input

```
name = input("Enter your name: ")
print("Hello", name)
```

## Try-Except

```
try:
```

# Python Exam Review

```
x = int(input("Enter a number: "))
except ValueError:
    print("Invalid input!")
```

## File Handling

```
f = open("data.txt", "r")
print(f.read())
f.close()
```

## File Write

```
f = open("data.txt", "w")
f.write("Hello file!")
f.close()
```

## Delete File

```
import os
os.remove("data.txt")
```

## Modules

```
import math
print(math.sqrt(16))
```

## With Statement

```
with open("data.txt", "r") as f:
    print(f.read())
```

## Classes & Objects

```
class Person:
    def __init__(self, name):
        self.name = name

    def greet(self):
        print("Hello", self.name)

p = Person("Alice")
p.greet()
```

## Inheritance

```
class Animal:
    def speak(self):
```

## Python Exam Review

```
print("Animal sound")

class Dog(Animal):
    def speak(self):
        print("Bark")

d = Dog()
d.speak()
```

### Polymorphism

```
class Cat:
    def sound(self):
        print("Meow")

class Dog:
    def sound(self):
        print("Bark")

for animal in (Cat(), Dog()):
    animal.sound()
```

### Encapsulation

```
class Car:
    def __init__(self):
        self.__speed = 0

    def set_speed(self, speed):
        self.__speed = speed

    def get_speed(self):
        return self.__speed

c = Car()
c.set_speed(100)
print(c.get_speed())
```