Python Comprehensive Task Document

1. Introduction

This document provides a comprehensive Python task covering all essential concepts from beginner to advanced level. It includes fundamental syntax, data structures, object-oriented programming, file handling, data analysis, visualization, and machine learning.

2. Basic Python Concepts

Topics covered:
- Variables, data types, and operators
- Conditional statements
- Loops
- Functions and recursion
- Exception handling

```
# Example
a = 10
b = 5
print("Sum:", a + b)

if a > b:
   print("A is greater")
else:
   print("B is greater")

def factorial(n):
   return 1 if n == 0 else n * factorial(n-1)

print("Factorial of 5:", factorial(5))
```

3. Python Data Structures

```
fruits = ["apple", "banana", "cherry"]
fruits.append("mango")
print("Updated list:", fruits)

scores = {"Alice": 90, "Bob": 85}
scores["Charlie"] = 88
print("Scores:", scores)

unique_nums = set([1, 2, 2, 3, 4])
print("Unique numbers:", unique_nums)
```

4. Object-Oriented Programming

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

    def show(self):
        print(f"{self.name} runs at {self.speed} km/h")

class Car(Vehicle):
    def show(self):
        print(f"Car {self.name} runs at {self.speed} km/h")

car1 = Car("Tesla", 120)
car1.show()
```

5. File Handling and Exception Management

```
try:
    with open("sample.txt", "w") as file:
        file.write("Hello, Python!")
    with open("sample.txt", "r") as file:
        print(file.read())
except Exception as e:
    print("Error:", e)
```

6. Data Analysis with Pandas and NumPy

```
import pandas as pd
import numpy as np

data = {"Name": ["A", "B", "C"], "Score": [85, 90, 78]}
df = pd.DataFrame(data)
print(df.describe())

arr = np.array([1, 2, 3, 4])
print("Squared:", np.square(arr))
```

7. Data Visualization

```
import matplotlib.pyplot as plt
import seaborn as sns

sns.set_style("darkgrid")
x = [1, 2, 3, 4]
y = [10, 20, 15, 25]
```

```
plt.plot(x, y, marker='o')
plt.title("Line Chart Example")
plt.show()
```

8. Machine Learning with Scikit-Learn

```
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import accuracy_score
from sklearn.datasets import load_iris

iris = load_iris()
X, y = iris.data, iris.target
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

model = LogisticRegression(max_iter=200)
model.fit(X_train, y_train)
preds = model.predict(X_test)
print("Accuracy:", accuracy_score(y_test, preds))
```

9. Advanced Topics

```
import requests, json

response = requests.get("https://api.github.com")
print("Status:", response.status_code)
print("Response JSON:", response.json())
```

10. Conclusion

This document provides structured code samples covering Python from A to Z. You can paste the provided snippets into Google Colab for execution and testing.