File Handling

```
# 12.2 - Working With File Paths in Python

# Solutions to review exercises



# Exercise 1

from pathlib import Path

file_path = Path.home() / "my_folder" / "my_file.txt"



# Exercise 2

print(file_path.exists())



# Exercise 3

print(file_path.name)



# Exercise 4

print(file_path.parent.name)

# 12.3 Common File System Operations

# Solutions to Exercises



# Exercise 1

from pathlib import Path
```

File Handling

```
new_dir = Path.home() / "my_folder"

new_dir.mkdir()



# Exercise 2

file1 = new_dir / "file1.txt"

file2 = new_dir / "file2.txt"

image1 = new_dir / "image1.png"


file1.touch()

file2.touch()

image1.touch()



# Exercise 3

images_dir = new_dir / "images"

images_dir.mkdir()

image1.replace(images_dir / "image1.png")



# Exercise 4

file1.unlink()



# Exercise 5

import shutil


shutil.rmtree(new_dir)
```

File Handling

```
# 12.4 Challenge: Move All Image Files To a New Directory

# Solution to Challenge


from pathlib import Path


# Change this path to match the location on your computer

documents_dir = Path.cwd() / "practice_files" / "documents"


# Create an images/ directory in your home directory

images_dir = Path.home() / "images"

images_dir.mkdir(exist_ok=True)


# Search for image files in the documents directory and move

# them to the images/ directory

for path in documents_dir.rglob("*.*"):

    if path.suffix.lower() in [".png", ".jpg", ".gif"]:

        path.replace(images_dir / path.name)


# 12.5 - Reading and Writing Files

# Solutions to Exercises



# Exercise 1

from pathlib import Path


starships = ["Discovery\n", "Enterprise\n", "Defiant\n", "Voyager"]


file_path = Path.home() / "starships.txt"

with file_path.open(mode="w", encoding="utf-8") as file:
```

File Handling

```
    file.writelines(starships)



# Exercise 2
with file_path.open(mode="r", encoding="utf-8") as file:
    for starship in file.readlines():
        print(starship, end="")



# Exercise 3
with file_path.open(mode="r", encoding="utf-8") as file:
    for starship in file.readlines():
        if starship.startswith("D"):
            print(starship, end="")

# 12.5 Read and Write CSV Data
# Solutions to Exercises



# Exercise 1
import csv
from pathlib import Path

numbers = [
    [1, 2, 3, 4, 5],
    [6, 7, 8, 9, 10],
    [11, 12, 13, 14, 15],
]
```

File Handling

```
file_path = Path.home() / "numbers.csv"

with file_path.open(mode="w", encoding="utf-8") as file:
    writer = csv.writer(file)
    writer.writerows(numbers)



# Exercise 2
numbers = []

with file_path.open(mode="r", encoding="utf-8") as file:
    reader = csv.reader(file)
    for row in reader:
        int_row = [int(num) for num in row]
        numbers.append(int_row)

print(numbers)



# Exercise 3
favorite_colors = [
    {"name": "Joe", "favorite_color": "blue"},
    {"name": "Anne", "favorite_color": "green"},
    {"name": "Bailey", "favorite_color": "red"},
]

file_path = Path.home() / "favorite_colors.csv"

with file_path.open(mode="w", encoding="utf-8") as file:
```

File Handling

```
    writer = csv.DictWriter(file, fieldnames=["name", "favorite_color"])
    writer.writeheader()
    writer.writerows(favorite_colors)



# Exercise 4
favorite_colors = []

with file_path.open(mode="r", encoding="utf-8") as file:
    reader = csv.DictReader(file)
    for row in reader:
        favorite_colors.append(row)

print(favorite_colors)

# 12.7 Challenge: Create a High Scores List
# Solution to Challenge

import csv
from pathlib import Path

# Change the path below to match the location on your computer
scores_csv_path = (
    Path.cwd()
    / "practice_files"
    / "scores.csv"
)

with scores_csv_path.open(mode="r", encoding="utf-8") as file:
```

File Handling

```
    reader = csv.DictReader(file)
    scores = [row for row in reader]


high_scores = {}
for item in scores:
    name = item["name"]
    score = int(item["score"])
    # If the name has not been added to the high_score dictionary, then
    # create a new key with the name and set its value to the score
    if name not in high_scores:
        high_scores[name] = score
    # Otherwise, check to see if score is greater than the score currently
    # assigned to high_scores[name] and replace it if it is
    else:
        if score > high_scores[name]:
            high_scores[name] = score

# The high_scores dictionary now contains one key for each name that was
# in the scores.csv file, and each value is that player's highest score.

output_csv_file = Path.cwd() / "high_scores.csv"
with output_csv_file.open(mode="w", encoding="utf-8") as file:
    writer = csv.DictWriter(file, fieldnames=["name", "high_score"])
    writer.writeheader()
    for name in high_scores:
        row_dict = {"name": name, "high_score": high_scores[name]}
        writer.writerow(row_dict)
```