File Handling — The Full Picture

// The pattern — always

```
with open("file.txt", "r",
          encoding="utf-8") as f:
    content = f.read()
# closed automatically
```

// Modes

```
"r"   # read — file must exist
"w"   # write — creates or overwrites
"a"   # append — adds to end
```

// Reading

```
f.read()          # whole file as string
f.readlines()     # all lines as list
f.readline()      # one line at a time
for line in f:    # line by line
    line.strip()  # removes \n
```

// Writing

```
f.write(string)   # add \n manually
f.writelines(lst) # add \n in strings
```

// Error handling

```
try:
    with open("f.txt", "r",
              encoding="utf-8") as f:
        content = f.read()
except FileNotFoundError:
    print("File not found.")
```

// Common mistakes

```
# no with open()   → file stays locked
# "w" mode         → silent overwrite
# missing \n       → one long line
# CSV values       → all strings
# no newline=""    → blank CSV rows
# read() twice     → second = ""
```

// CSV — reading

```
import csv

# rows as lists
with open("f.csv","r",encoding="utf-8") as f:
    reader = csv.reader(f)
    next(reader)       # skip header
    for row in reader:
        print(row)     # list of strings

# rows as dicts
with open("f.csv","r",encoding="utf-8") as f:
    reader = csv.DictReader(f)
    for row in reader:
        print(row["key"])
```

// CSV — writing

```
# write lists
with open("f.csv","w",encoding="utf-8",
          newline="") as f:
    writer = csv.writer(f)
    writer.writerow(["name","score"])
    writer.writerows(rows)

# write dicts
with open("f.csv","w",encoding="utf-8",
          newline="") as f:
    writer = csv.DictWriter(f,
        fieldnames=["name","score"])
    writer.writeheader()
    writer.writerows(dicts)
```

// CSV — all values are strings

```
int(row["score"])      # convert to int
float(row["price"])    # convert to float
row["active"]=="True"  # convert to bool
```