Tutorial

A Brief Introduction to Computational Tools

Serhat Beyenir

11 November 2025

Contents

List of Figures

List of Tables

Preface

A Brief Introduction to Computational Tools presents a collection of tutorials based on lecture notes from classes, designed to give learners clear and essential insights into key topics. No previous programming experience is required. Each tutorial guides you step by step through the concepts with hands-on examples.

Chapter 1

Python Tutorial

This tutorial introduces Python programming, covering basic concepts with examples to illustrate key points. We will start by using Python as a calculator, then explore variables, functions, and control flow.

1.1 Requirements

To follow this tutorial, the easiest way to get started is by using a web-based Python environment. This lets you write and run Python code right in your browser; no downloads or setup needed. I recommend python-fiddle.com, an easy-to-use online editor that lets you experiment with Python instantly and solve your problem sets effortlessly.

If you prefer working on your own computer, make sure you have Python(version 3.10 or later) installed. Python works on Windows, macOS, and Linux. You'll also need a text editor or an Integrated Development Environment (IDE) to write your code. I recommend Positron, a beginner-friendly IDE with a built-in terminal, though other editors like VS Code or PyCharm are also great options.

1.2 Basic Syntax

Python uses indentation (typically four spaces) to define code blocks. A colon (:) introduces a block, and statements within the block must be indented consistently. Python is case-sensitive, so `Variable` and `variable` are distinct identifiers. *Statements typically end with a newline, but you can use a backslash (\) to continue a statement across multiple lines.*

```python
total = 1 + 2 + 3 + 4 + 5
print(total)  # Output: 15
```

Basic syntax rules:

- Comments start with `#` and extend to the end of the line.
- Strings can be enclosed in single quotes (`'`), double quotes (`"`), or triple quotes (`'''` or `"""`) for multi-line strings.
- Python is case-sensitive, so `Variable` and `variable` are different identifiers.

1.3 The `print()` Function

The `print()` function displays output in Python.

```python
name = "Rudolf Diesel"
year = 1858
print(f"{name} was born in {year}.")
```

Output: `Rudolf Diesel was born in 1858.`

1.4 Formatting in `print()`

The following table illustrates common f-string formatting options for the `print()` function:

Format	Code	Example	Output
Round to 2 decimals	`f"{x:.2f}"`	`print(f"{3.14159:.2f}")`	`3.14`
Round to whole number	`f"{x:.0f}"`	`print(f"{3.9:.0f}")`	`4`
Thousands separator	`f"{x:,.2f}"`	`print(f"{1234567.89:,.2f}")`	`1,234,567.89`
Percentage	`f"{x:.1%}"`	`print(f"{0.756:.1%}")`	`75.6%`
Currency style	`f"${x:,.2f}"`	`print(f"${1234.5:,.2f}")`	`$1,234.50`

Note: The currency symbol (e.g., `$`) can be modified for other currencies (e.g., `€`, `£`) based on the desired locale.

1.5 Variables and Data Types

Variables store data and are assigned values using the `=` operator.

```python
x = 10
y = 3.14
name = "Rudolph"
```

Python has several built-in data types, including:

- Integers (`int`): Whole numbers, e.g., `10`, `-5`

- Floating-point numbers (`float`): Decimal numbers, e.g., `3.14`, `-0.001`
- Strings (`str`): Text, e.g., `"Hello"`, `'World'`
- Booleans (`bool`): `True` or `False`

1.5.1 Arithmetic Operations

```
a = 10
b = 3
print(a + b)  # Addition: 13
print(a - b)  # Subtraction: 7
print(a * b)  # Multiplication: 30
print(a / b)  # Division: 3.3333...
print(a // b) # Integer Division: 3
print(a ** b) # Exponentiation: 1000
```

1.5.2 String Operations

```
first_name = "Rudolph"
last_name = "Diesel"
full_name = first_name + " " + last_name  # Concatenation using +
print(full_name)  # Output: Rudolph Diesel
print(f"{first_name} {last_name}")  # Concatenation using f-string
print(full_name * 2)  # Repetition: Rudolph DieselRudolph Diesel
print(full_name.upper())  # Uppercase: RUDOLPH DIESEL
```

Note: String repetition (`*`) concatenates the string multiple times without spaces. For example, `full_name * 2` produces `Rudolph DieselRudolph Diesel`.

1.6 Python as a Calculator in Interactive Mode

Python's interactive mode allows you to enter commands and see results immediately, ideal for quick calculations. To start, open a terminal (on macOS, Linux, or Windows) and type:

```
python3  # Use 'python' on Windows if 'python3' is not recognized
```

You should see the Python prompt:

```
>>>
```

Enter expressions and press **Enter** to see results:

```
2 + 3  # Output: 5
7 - 4  # Output: 3
6 * 9  # Output: 54
8 / 2  # Output: 4.0
```

```
8 // 2 # Output: 4
2 ** 3 # Output: 8
```

1.6.1 Parentheses for Grouping

```
(2 + 3) * 4  # Output: 20
2 + (3 * 4)  # Output: 14
```

1.6.2 Variables

```
x = 10
y = 3
x / y  # Output: 3.3333333333333335
```

1.6.3 Exiting Interactive Mode

To exit, type:

```
exit()
```

Alternatively, use: - **Ctrl+D** (macOS/Linux) - **Ctrl+Z** then Enter (Windows)

1.7 Control Flow

Control flow statements direct the execution of code based on conditions.

1.7.1 Conditional Statements

Conditional statements allow you to execute different code blocks based on specific conditions. Python provides three keywords for this purpose:

- **`if`**: Evaluates a condition and executes its code block if the condition is `True`.
- **`elif`**: Short for "else if," it checks an additional condition if the preceding `if` or `elif` conditions are `False`. You can use multiple `elif` statements to test multiple conditions sequentially, and Python will execute the first `True` condition's block, skipping the rest.
- **`else`**: Executes a code block if none of the preceding `if` or `elif` conditions are `True`. It serves as a fallback and does not require a condition.

The following example uses age to categorize a person as a Minor, Adult, or Senior, demonstrating how `if`, `elif`, and `else` work together.

```
# Categorize a person based on their age
age = 19
if age < 18:
```

```
    print("Minor")
elif age <= 64:
    print("Adult")
else:
    print("Senior")
```

Output: `Adult`

1.7.2 For Loop

A `for` loop iterates over a sequence (e.g., list or string).

```
components = ["piston", "liner", "connecting rod"]
for component in components:
    print(component)
```

Output:

```
piston
liner
connecting rod
```

1.7.3 While Loop

A `while` loop executes as long as a condition is true. Ensure the condition eventually becomes false to avoid infinite loops.

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

Output:

```
0
1
2
3
4
5
```

1.8 Functions

1.8.1 The `def` Keyword

Functions are reusable code blocks defined using the `def` keyword. They can include default parameters for optional arguments.

```
def add(a, b=0):
    return a + b
print(add(5))      # Output: 5
print(add(5, 3))   # Output: 8

def multiply(*args):
    result = 1
    for num in args:
        result *= num
    return result
print(multiply(2, 3, 4))  # Output: 24
```

1.8.2 The `lambda` Keyword

The `lambda` keyword creates anonymous functions for short, one-off operations, often used in functional programming.

```
celsius_to_fahrenheit = lambda c: (c * 9 / 5) + 32
print(celsius_to_fahrenheit(25))  # Output: 77.0
```

1.9 The `math` Module

The `math` module provides mathematical functions and constants.

```
import math
print(math.sqrt(16))  # Output: 4.0
print(math.pi)        # Output: 3.141592653589793
```

```
import math
angle = math.pi / 4  # 45 degrees in radians
print(math.sin(angle))  # Output: 0.7071067811865475 (approximately √2/2)
print(math.cos(angle))  # Output: 0.7071067811865476 (approximately √2/2)
print(math.tan(angle))  # Output: 1.0
```

Note: Floating-point arithmetic may result in small precision differences, as seen in the `sin` and `cos` outputs.

```
import math
print(math.log(10))       # Natural logarithm of 10: 2.302585092994046
print(math.log(100, 10))  # Logarithm of 100 with base 10: 2.0
```

1.9.1 Converting Between Radians and Degrees

The `math` module provides `math.radians()` to convert degrees to radians and `math.degrees()` to convert radians to degrees, which is useful for trigonometric calculations.

```
import math
degrees = 180
radians = math.radians(degrees)
print(f"{degrees} degrees is {radians:.3f} radians")  # Output: 180 degrees is 3.142 radians

radians = math.pi / 2
degrees = math.degrees(radians)
print(f"{radians:.3f} radians is {degrees:.1f} degrees")  # Output: 1.571 radians is 90.0 degrees
```

1.10 Writing Python Scripts

Write Python code in a `.py` file and run it as a script. Create a file named `script.py`:

```
# script.py
import math
print("Square root of 16 is:", math.sqrt(16))
print("Value of pi is:", math.pi)
print("Sine of 90 degrees is:", math.sin(math.pi / 2))
print("Natural logarithm of 10 is:", math.log(10))
print("Logarithm of 100 with base 10 is:", math.log(100, 10))
```

To run the script, open a terminal, navigate to the directory containing `script.py` using the `cd` command (e.g., `cd /path/to/directory`), and type:

```
python3 script.py  # or python script.py on Windows
```

Output:

```
Square root of 16 is: 4.0
Value of pi is: 3.141592653589793
Sine of 90 degrees is: 1.0
Natural logarithm of 10 is: 2.302585092994046
Logarithm of 100 with base 10 is: 2.0
```

1.11 Summary

This tutorial covered Python basics, including syntax, variables, data types, operations, control flow, and functions. Python's rich ecosystem includes libraries like:

- **NumPy**: For numerical computations and array manipulations.
- **Matplotlib**: For data visualization and plotting.
- **Pandas**: For data manipulation and analysis with tabular data structures.
- **Pint**: For handling physical quantities and performing unit conversions.

You can explore these libraries to enhance your Python programming skills further. For example installing them can be done using `pip`:

```
pip install numpy matplotlib pandas pint
```

`pip` is Python's package manager for installing and managing additional libraries.

Chapter 2

SI Units

The International System of Units (SI) is the globally accepted standard for measurement. Established to provide a consistent framework for scientific and technical measurements, SI units facilitate clear communication and data comparison across various fields and countries. The system is based on seven fundamental units: the meter for length, the kilogram for mass, the second for time, the ampere for electric current, the kelvin for temperature, the mole for substance, and the candela for luminous intensity.

Table 2.1: Base SI units.

Physical Quantity	SI Base Unit	Symbol
Length	Meter	m
Mass	Kilogram	kg
Time	Second	s
Electric Current	Ampere	A
Temperature	Kelvin	K
Amount of Substance	Mole	mol
Luminous Intensity	Candela	cd

Table 2.2: Derived SI units.

Physical Quantity	Derived SI Unit	Symbol
Area	Square meter	m^2
Volume	Cubic meter	m^3
Speed	Meter per second	m/s
Acceleration	Meter per second squared	m/s^2
Force	Newton	N
Pressure	Pascal	Pa
Energy	Joule	J
Power	Watt	W
Electric Charge	Coulomb	C
Electric Potential	Volt	V
Resistance	Ohm	Ω
Capacitance	Farad	F
Frequency	Hertz	Hz
Luminous Flux	Lumen	lm
Illuminance	Lux	lx
Specific Energy	Joule per kilogram	J/kg
Specific Heat Capacity	Joule per kilogram Kelvin	$J/(kg \cdot K)$

Table 2.3: Common multiples and submultiples for SI units.

Factor	Prefix	Symbol
10^9	giga	G
10^6	mega	M
10^3	kilo	k
10^2	hecto	h
10^1	deca	da
10^{-1}	deci	d
10^{-2}	centi	c
10^{-3}	milli	m
10^{-6}	micro	µ

2.1 Metric Multipliers for Length, Area, and Volume

The metric system uses powers of ten to scale measurements. When converting **area** or **volume**, remember that the conversion factor must be **squared** or **cubed**, respectively.

Unit	Symbol	Multiplier (to metres)	1 m = ?	Area (1 m² = ?)	Volume (1 m³ = ?)
Millimetre	mm	(10^{-3})	1000 mm	$(1000)^2 = 10^6\text{mm}^2$	$(1000)^3 = 10^9\text{mm}^3$
Centimetre	cm	(10^{-2})	100 cm	$(100)^2 = 10^4\text{cm}^2$	$(100)^3 = 10^6\text{cm}^3$
Decimetre	dm	(10^{-1})	10 dm	$(10)^2 = 10^2\text{dm}^2$	$(10)^3 = 10^3\text{dm}^3$
Metre	m	(10^0)	—	—	—
Decametre	dam	(10^1)	0.1 dam	$(0.1)^2 = 10^{-2}\text{dam}^2$	$(0.1)^3 = 10^{-3}\text{dam}^3$
Hectometre	hm	(10^2)	0.01 hm	$(0.01)^2 = 10^{-4}\text{hm}^2$	$(0.01)^3 = 10^{-6}\text{hm}^3$
Kilometre	km	(10^3)	0.001 km	$(0.001)^2 = 10^{-6}\text{km}^2$	$(0.001)^3 = 10^{-9}\text{km}^3$

2.1.1 Quick Reference

Conversion Type	Relationship
Length	1 m = 10^3 mm = 10^2 cm
Area	1 m^2 = 10^6 mm^2 = 10^4 cm^2
Volume	1 m^3 = 10^9 mm^3 = 10^6 cm^3

Note: When converting between metric units:

- Multiply by 10^n when going to a smaller unit.
- Divide by 10^n when going to a larger unit.
- For **area**, square the length factor; for **volume**, cube it.

2.2 Unity Fraction

The **unity fraction** method, or **unit conversion using unity fractions**, is a systematic way to convert one unit of measurement into another. This method relies on multiplying by fractions that are equal to one, where the numerator and the denominator represent the same quantity in different units. Since any number multiplied by one remains the same, unity fractions allow for seamless conversion without changing the value.

The principle of unity fractions is based on:

1. **Setting up equal values**: Write a fraction where the numerator and denominator are equivalent values in different units, so the fraction equals one. For example, $\frac{1km}{1000m}$ is a unity fraction because 1 km equals 1000 m.
2. **Multiplying by unity fractions**: Multiply the initial quantity by the unity fraction(s) so that the undesired units cancel out, leaving only the desired units.

2.2.1 Classwork

Example 2.1. Suppose we want to convert 5 kilometers to meters.

1. Start with 5 kilometers:
$$5\,\mathrm{km}$$
2. Multiply by a unity fraction that cancels kilometers and introduces meters. We use $(\frac{1000\,\mathrm{m}}{1\,\mathrm{km}}), since\, 1\,\mathrm{km} = 1000\,\mathrm{m}$:
$$5\,\mathrm{km} \times \frac{1000\,\mathrm{m}}{1\,\mathrm{km}} = 5000\,\mathrm{m}$$
3. The kilometers km cancel out, leaving us with meters m:

$$5\,\text{km} = 5000\,\text{m}$$

This step-by-step approach illustrates how the unity fraction cancels the undesired units and achieves the correct result in meters.

Unity fractions can be extended by using multiple conversion steps. For example, converting hours to seconds would require two unity fractions: one to convert hours to minutes and another to convert minutes to seconds. This approach ensures accuracy and is widely used in science, engineering, and other fields that require precise unit conversions.

Example 2.2. Convert 15 m/s to km/h.

1. Start with 15 m/s.
2. To convert meters to kilometers, multiply by $\frac{1\,\text{km}}{1000\,\text{m}}$.
3. To convert seconds to hours, multiply by $\frac{3600\,\text{s}}{1\,\text{h}}$.

$$15\,\text{m/s} \times \frac{1\,\text{km}}{1000\,\text{m}} \times \frac{3600\,\text{s}}{1\,\text{h}} = 54\,\text{km/h}$$

The meters and seconds cancel out, leaving kilometers per hour: 54 km/h.

2.2.2 Self-Test

Instructions:

1. Use unity fraction to convert between derived SI units.
2. Show each step of your work to ensure accuracy.
3. Simplify your answers and include correct units.

---

1. **Speed**
   Convert 72 km/h to m/s.
2. **Force**
   Convert 980 N (newtons) to $\text{kg} \cdot \text{m/s}^2$.
3. **Energy**
   Convert 2500 J (joules) to kJ.
4. **Power**
   Convert 1500 W (watts) to kW.
5. **Pressure**
   Convert 101325 Pa (pascals) to kPa.
6. **Volume Flow Rate**
   Convert $3\,\text{m}^3$/min to L/s.

7. **Density**
   Convert $1000\,\mathrm{kg/m^3}$ to $\mathrm{g/cm^3}$.
8. **Acceleration**
   Convert $9.8\,\mathrm{m/s^2}$ to $\mathrm{cm/s^2}$.
9. **Torque**
   Convert $50\,\mathrm{N\cdot m}$ to $\mathrm{kN\cdot cm}$.
10. **Frequency**
    Convert $500\,\mathrm{Hz}$ (hertz) to kHz.
11. **Work to Energy Conversion**
    A force of $20\,\mathrm{N}$ moves an object $500\,\mathrm{cm}$. Convert the work done to joules.
12. **Kinetic Energy Conversion**
    Calculate the kinetic energy in kilojoules of a $1500\,\mathrm{kg}$ car moving at $72\,\mathrm{km/h}$.
13. **Power to Energy Conversion**
    A machine operates at $2\,\mathrm{kW}$ for 3 hours. Convert the energy used to megajoules.
14. **Pressure to Force Conversion**
    Convert a pressure of $200\,\mathrm{kPa}$ applied to an area of $0.5\,\mathrm{m^2}$ to force in newtons.
15. **Density to Mass Conversion**
    Convert $0.8\,\mathrm{g/cm^3}$ for an object with a volume of $250\,\mathrm{cm^3}$ to mass in grams.

2.2.3 Answer Key

1. $72\,\mathrm{km/h} = 20\,\mathrm{m/s}$
2. $980\,\mathrm{N} = 980\,\mathrm{kg\cdot m/s^2}$
3. $2500\,\mathrm{J} = 2.5\,\mathrm{kJ}$
4. $1500\,\mathrm{W} = 1.5\,\mathrm{kW}$
5. $101325\,\mathrm{Pa} = 101.325\,\mathrm{kPa}$
6. $3\,\mathrm{m^3/min} = 50\,\mathrm{L/s}$
7. $1000\,\mathrm{kg/m^3} = 1\,\mathrm{g/cm^3}$
8. $9.8\,\mathrm{m/s^2} = 980\,\mathrm{cm/s^2}$
9. $50\,\mathrm{N\cdot m} = 5\,\mathrm{kN\cdot cm}$
10. $500\,\mathrm{Hz} = 0.5\,\mathrm{kHz}$
11. $20\,\mathrm{N} \times 5\,\mathrm{m} = 100\,\mathrm{J}$
12. Kinetic energy $= 1500\,\mathrm{kg} \times (20\,\mathrm{m/s})^2/2 = 300\,\mathrm{kJ}$
13. $2\,\mathrm{kW} \times 3\,\mathrm{hours} = 21.6\,\mathrm{MJ}$
14. $200\,\mathrm{kPa} \times 0.5\,\mathrm{m^2} = 100,000\,\mathrm{N}$
15. $0.8\,\mathrm{g/cm^3} \times 250\,\mathrm{cm^3} = 200\,\mathrm{g}$

2.3 Condenser Vacuum

Condenser vacuum gauge reads 715 mmHg when barometer stands at 757 mmHg. State the absolute pressure in kN/m² and bar.

2.3.1 Given Data

$$P_{atm} = 757 \text{ mmHg}, \quad P_{vac} = 715 \text{ mmHg}$$

2.3.2 Absolute Pressure in mmHg

$$P_{abs} = P_{atm} - P_{vac} = 757 - 715 = 42 \text{ mmHg}$$

2.3.3 Convert mmHg → kN/m²

$$P = \rho g h = 13{,}600 \cdot 9.81 \cdot 0.001 = 133.416 \text{ Pa} = 0.133416 \text{ kN/m}^2$$

$$P_{abs} = 42 \cdot 0.133416 = 5.6034 \text{ kN/m}^2$$

2.3.4 Convert kN/m² → bar

$$P_{abs} = \frac{5.6034}{100} = 0.056 \text{ bar}$$

2.3.5 Final Answers

$$\boxed{P_{abs} = 42 \text{ mmHg} = 5.6034 \text{ kN/m}^2 = 0.056 \text{ bar}}$$

2.3.6 Code

```
P_atm_mmHg = 757
P_vac_mmHg = 715
MMHG_TO_KN_M2 = 0.133416
KNM2_TO_BAR = 1 / 100
P_abs_mmHg = P_atm_mmHg - P_vac_mmHg
P_abs_kNm2 = P_abs_mmHg * MMHG_TO_KN_M2
P_abs_bar = P_abs_kNm2 * KNM2_TO_BAR
print(f"Absolute Pressure = {P_abs_mmHg:.2f} mmHg")
print(f"Absolute Pressure = {P_abs_kNm2:.3f} kN/m²")
print(f"Absolute Pressure = {P_abs_bar:.4f} bar")
```

2.4 Oil Flow in Tubes

Oil flows full bore at a velocity of $V = 2$ m/s through 16 tubes of diameter $d = 30$ mm. Density of oil: $\rho = 0.85$ g/mL. Find **volume flow rate** (L/s) and **mass flow rate** (kg/min).

2.4.1 Cross-sectional area of one tube

$$A = \pi \frac{d^2}{4} = \pi \frac{0.03^2}{4} \approx 7.0686 \times 10^{-4} \text{ m}^2$$

2.4.2 Total area and volume flow rate

$$A_{\text{total}} = 16 \cdot 7.0686 \times 10^{-4} \approx 0.01131 \text{ m}^2$$

$$\dot{v} = A_{\text{total}} \cdot V \approx 0.02262 \text{ m}^3/\text{s} \approx 22.62 \text{ L/s}$$

2.4.3 Mass flow rate

$$\dot{m} = \rho \cdot \dot{v} = 850 \cdot 0.02262 \approx 19.227 \text{ kg/s} \approx 1153.6 \text{ kg/min}$$

2.4.4 Final Answers

$$\text{Volume flow rate: } \dot{v} \approx 22.6 \text{ L/s}$$

$$\text{Mass flow rate: } \dot{m} \approx 1154 \text{ kg/min}$$

2.4.5 Code

```python
import math
v = 2.0
N = 16
d = 0.03
rho = 0.85 * 1000
A = math.pi * d**2 / 4
A_total = N * A
v_dot_m3_s = A_total * v
v_dot_L_s = v_dot_m3_s * 1000
m_dot_kg_s = rho * v_dot_m3_s
m_dot_kg_min = m_dot_kg_s * 60
print(f"Volume flow rate: {v_dot_L_s:.2f} L/s")
print(f"Mass flow rate: {m_dot_kg_min:.2f} kg/min")
```

2.5 Gauge Pressure

An oil of specific gravity (relative density) SG = 0.8 is contained in a vessel to a depth of $h = 2$ m. Find the **gauge pressure** at this depth in kPa.

2.5.1 Gauge Pressure

$$P_g = \rho g h$$

where

ρ = density of fluid (kg/m^3)

g= acceleration due to gravity (9.81 m/s^2)

h = depth (m)

2.5.2 Compute the density of oil using specific gravity

Specific gravity is defined as

$$\mathrm{SG} = \frac{\rho_{\mathrm{oil}}}{\rho_{\mathrm{water}}}$$

where $\rho_{\mathrm{water}} = 1000\ \ \mathrm{kg/m^3}$. Thus,

$$\rho_{\mathrm{oil}} = \mathrm{SG} \times \rho_{\mathrm{water}} = 0.8 \times 1000 = 800\ \mathrm{kg/m^3}$$

2.5.3 Compute the gauge pressure

$$P_g = \rho g h = 800 \times 9.81 \times 2$$

$$P_g = 15696\ \mathrm{Pa} \approx 15.7\ \mathrm{kPa}$$

2.5.4 Answer

The **gauge pressure** at a depth of 2 m in the oil is:

$$\boxed{15.7\ \mathrm{kPa}}$$

2.5.5 Code

```
# Gauge Pressure Calculation for Oil

# Given data
specific_gravity = 0.8  # SG of oil
depth_m = 2.0           # depth in meters
g = 9.81                # acceleration due to gravity in m/s²
rho_water = 1000        # density of water in kg/m³

# Compute density of oil
rho_oil = specific_gravity * rho_water

# Compute gauge pressure (Pa)
P_g_Pa = rho_oil * g * depth_m

# Convert to kPa
P_g_kPa = P_g_Pa / 1000

# Print results
print(f"Density of oil: {rho_oil:.1f} kg/m³")
print(f"Gauge pressure at {depth_m} m depth: {P_g_Pa:.1f} Pa ({P_g_kPa:.2f} kPa)")
```

2.6 Absolute Pressure from Manometer Reading

A water manometer shows a pressure in a vessel of 400 mm **below atmospheric pressure**. The atmospheric pressure is measured as 763 mmHg. Determine the **absolute pressure** in the vessel in kPa.

2.6.1 Relationship between absolute and gauge pressure

$$P_{\text{abs}} = P_{\text{atm}} + P_{\text{gauge}}$$

Since the manometer shows a pressure **below atmospheric**, the gauge pressure is negative:

$$P_{\text{gauge}} = -\rho_{\text{water}} g h$$

2.6.2 Convert atmospheric pressure to Pa using

$$P = \rho g h = 13{,}600 \cdot 9.81 \cdot 0.001 = 133.416 \text{ Pa}$$

So

$$P_{\text{atm}} = 763 \times 133.416 \approx 101,801 \text{ Pa} \approx 101.8 \text{ kPa}$$

2.6.3 Compute gauge pressure

Water column height:

$$h = 400 \text{ mm} = 0.4 \text{ m}$$

Density of water: $rho_{\text{water}} = 1000 \text{ kg/m}^3$, $g = 9.81 \text{ m/s}^2$

$$P_{\text{gauge}} = -\rho g h = -1000 \times 9.81 \times 0.4$$

$$P_{\text{gauge}} = -3924 \text{ Pa} \approx -3.92 \text{ kPa}$$

2.6.4 Compute absolute pressure

$$P_{\text{abs}} = P_{\text{atm}} + P_{\text{gauge}} \approx 101.8 - 3.92 \approx 97.9 \text{ kPa}$$

2.6.5 Answer

The **absolute pressure** in the vessel is:

$$\boxed{97.9 \text{ kPa}}$$

2.6.6 Code

```
# Absolute Pressure Calculation from Water Manometer

# Given data
h_mm = 400                 # manometer reading in mm (below atmospheric)
atm_mmHg = 763             # atmospheric pressure in mmHg
rho_water = 1000           # density of water in kg/m³
g = 9.81                   # gravity in m/s²
mmHg_to_Pa = 133.416       # conversion factor

# Convert manometer height to meters
h_m = h_mm / 1000

# Convert atmospheric pressure to Pa
P_atm_Pa = atm_mmHg * mmHg_to_Pa

# Gauge pressure (negative because below atmospheric)
P_gauge_Pa = - rho_water * g * h_m

# Absolute pressure
```

```
P_abs_Pa = P_atm_Pa + P_gauge_Pa

# Convert to kPa
P_abs_kPa = P_abs_Pa / 1000

# Print results
print(f"Atmospheric pressure: {P_atm_Pa:.1f} Pa ({P_atm_Pa/1000:.1f} kPa)")
print(f"Gauge pressure: {P_gauge_Pa:.1f} Pa ({P_gauge_Pa/1000:.2f} kPa)")
print(f"Absolute pressure in the vessel: {P_abs_Pa:.1f} Pa ({P_abs_kPa:.2f} kPa)")
```

Chapter 3

Heat and Work

3.1 Heat Required to Heat Steel

A steel block of mass $m = 5$ kg and specific heat capacity $c = 480$ J/kg $\cdot$ K is heated from $T_1 = 15°$C to $T_2 = 100°$C. Determine the **heat required**.

3.1.1 The formula for heat

$$Q = mc\Delta T \tag{3.1}$$

where

$\Delta T = T_2 - T_1$ is the temperature change.

3.1.2 Compute the temperature change

$$\Delta T = T_2 - T_1 = 100 - 15 = 85 \text{ K}$$

3.1.3 Compute the heat required

$$Q = mc\Delta T = 5 \times 480 \times 85$$

$$Q = 204,000 \text{ J}$$

3.1.4 Answer

The **heat required** to raise the temperature of the steel is:

$$\boxed{204 \text{ kJ}}$$

3.1.5 Code

```
# Heat Required to Heat Steel

# Given data
mass = 5            # kg
specific_heat = 480 # J/kg·K
T_initial = 15      # °C
T_final = 100       # °C

# Temperature change
delta_T = T_final - T_initial

# Heat required (in J)
Q_J = mass * specific_heat * delta_T

# Convert to kJ
Q_kJ = Q_J / 1000

# Print results
print(f"Temperature change: {delta_T} K")
print(f"Heat required: {Q_J:.0f} J ({Q_kJ:.0f} kJ)")
```

```
# Interactive Heat Calculation

# Get user input
mass = float(input("Enter the mass of the object (kg): "))
specific_heat = float(input("Enter the specific heat capacity (J/kg·K): "))
T_initial = float(input("Enter the initial temperature (°C): "))
T_final = float(input("Enter the final temperature (°C): "))

# Calculate temperature change
delta_T = T_final - T_initial

# Calculate heat required
Q_J = mass * specific_heat * delta_T
Q_kJ = Q_J / 1000

# Display results
print("\nCalculation Results:")
print(f"Temperature change: {delta_T:.2f} K")
print(f"Heat required: {Q_J:.2f} J ({Q_kJ:.2f} kJ)")
```

3.2 Finding Specific Heat Capacity

A liquid of mass $m = 4$ kg is heated from $T_1 = 15°\text{C}$ to $T_2 = 100°\text{C}$. The heat supplied is $Q = 714$ kJ. Determine the **specific heat capacity** c of the liquid.

3.2.1 Recall the formula for heat

$$Q = mc\Delta T$$

where $\Delta T = T_2 - T_1$.

3.2.2 Convert heat to joules

$$Q = 714 \text{ kJ} = 714 \times 1000 = 714,000 \text{ J}$$

3.2.3 Compute temperature change

$$\Delta T = T_2 - T_1 = 100 - 15 = 85 \text{ K}$$

3.2.4 Solve for specific heat capacity

$$c = \frac{Q}{m\Delta T} = \frac{714,000}{4 \times 85}$$

$$c = \frac{714,000}{340} \approx 2100 \text{ J/kg} \cdot \text{K}$$

3.2.5 Answer

The **specific heat capacity** of the liquid is:

$$\boxed{c \approx 2100 \text{ J/kg} \cdot \text{K}}$$

3.2.6 Code

```
# Specific Heat Capacity Calculation

# Given data
mass = 4            # kg
T_initial = 15      # °C
T_final = 100       # °C
Q_kJ = 714          # heat supplied in kJ

# Convert heat to joules
Q_J = Q_kJ * 1000
```

```
# Temperature change
delta_T = T_final - T_initial

# Calculate specific heat capacity
c = Q_J / (mass * delta_T)

# Print results
print(f"Temperature change: {delta_T} K")
print(f"Specific heat capacity: {c:.0f} J/kg·K")
```

```
# Interactive Specific Heat Capacity Calculator

# Get user input
mass = float(input("Enter the mass of the liquid (kg): "))
T_initial = float(input("Enter the initial temperature (°C): "))
T_final = float(input("Enter the final temperature (°C): "))
Q_kJ = float(input("Enter the heat supplied (kJ): "))

# Convert heat to joules
Q_J = Q_kJ * 1000

# Temperature change
delta_T = T_final - T_initial

# Calculate specific heat capacity
c = Q_J / (mass * delta_T)

# Display results
print("\nCalculation Results:")
print(f"Temperature change: {delta_T:.2f} K")
print(f"Specific heat capacity: {c:.2f} J/kg·K")
```

3.3 Work Done by Fluid in Cylinder

A fluid in a cylinder is at pressure $P = 700$ kPa. It is **expanded at constant pressure** from a volume of $V_1 = 0.28\ \text{m}^3$ to $V_2 = 1.68\ \text{m}^3$. Determine the **work done**.

3.3.1 The formula for work done at constant pressure

$$W = P\Delta V \tag{3.2}$$

where

$$\Delta V = V_2 - V_1$$

3.3.2 Compute the change in volume

$$\Delta V = 1.68 - 0.28 = 1.40 \text{ m}^3$$

3.3.3 Convert pressure to Pa

$$P = 700 \text{ kPa} = 700 \times 10^3 \text{ Pa} = 700,000 \text{ Pa}$$

3.3.4 Compute the work done

$$W = P\Delta V = 700,000 \times 1.40$$

$$W = 980,000 \text{ J} \approx 980 \text{ kJ}$$

3.3.5 Answer

The **work done** by the fluid during expansion is:

$$\boxed{980 \text{ kJ}}$$

3.3.6 Code

```
# Work Done by Fluid at Constant Pressure

# Given data
P_kPa = 700          # pressure in kPa
V1 = 0.28            # initial volume in m³
V2 = 1.68            # final volume in m³

# Convert pressure to Pa
P_Pa = P_kPa * 1000

# Compute change in volume
delta_V = V2 - V1

# Compute work done (J)
W_J = P_Pa * delta_V

# Convert to kJ
W_kJ = W_J / 1000

# Print results
print(f"Pressure: {P_Pa} Pa")
print(f"Change in volume: {delta_V:.2f} m³")
print(f"Work done: {W_J:.0f} J ({W_kJ:.0f} kJ)")
```

```
# Interactive Work Done Calculator

# Get user input
P_kPa = float(input("Enter the constant pressure (kPa): "))
V1 = float(input("Enter the initial volume (m³): "))
V2 = float(input("Enter the final volume (m³): "))

# Convert pressure to Pa
P_Pa = P_kPa * 1000

# Compute change in volume
delta_V = V2 - V1

# Compute work done (J)
W_J = P_Pa * delta_V

# Convert to kJ
W_kJ = W_J / 1000

# Display results
print("\nCalculation Results:")
print(f"Pressure: {P_Pa:.0f} Pa ({P_kPa:.0f} kPa)")
print(f"Change in volume: {delta_V:.2f} m³")
print(f"Work done: {W_J:.0f} J ({W_kJ:.2f} kJ)")
```

3.4 Enthalpy Calculation

120 kg of steam at $p = 1000$ kPa and $T = 200°$C has internal energy $u = 2623$ kJ/kg and specific volume ($v = 0.2061\ \ \mathrm{m}^3$/kg Calculate the specific enthalpy (h) and the total enthalpy for the 120-kg mass.

Given for steam:

- mass $m = 120\ \ \mathrm{kg}$
- pressure $P = 1000\ \ \mathrm{kPa}$
- temperature $T = 200°\mathrm{C}$ (not needed for calculation)
- internal energy $u = 2623\ \ \mathrm{kJ/kg}$
- specific volume $v = 0.2061\ \ \mathrm{m}^3/\mathrm{kg}$

The specific enthalpy is

$$h = u + Pv \tag{3.3}$$

Note: with (P) in kPa and (v) in m(^3)/kg, the product (Pv) has units kJ/kg (since (1 kPa m^3 = 1 kJ)).

Compute (Pv):

$$Pv = 1000 \times 0.2061 = 206.1 \text{ kJ/kg}$$

So the specific enthalpy is:

$$h = 2623 + 206.1 = 2829.1 \text{ kJ/kg}$$

Total enthalpy for the 120 kg of steam:

$$H = m\,h = 120 \times 2829.1 = 339{,}492 \text{ kJ}$$

3.4.1 Answers

- Specific enthalpy: $\boxed{h = 2829.1 \text{ kJ/kg}}$
- Total enthalpy for 120 kg: $\boxed{H = 339{,}492 \text{ kJ}}$

3.4.2 Code

```python
# Given data
m = 120           # kg
P = 1000          # kPa
u = 2623          # kJ/kg
v = 0.2061        # m³/kg

# Enthalpy calculation
h = u + P * v           # kJ/kg
H_total = m * h         # kJ

print(f"Specific enthalpy h = {h:.1f} kJ/kg")
print(f"Total enthalpy H = {H_total:,.0f} kJ")
```

Chapter 4

Thermal Expansion

Thermal expansion is the tendency of materials to change their shape, area, and volume in response to a change in temperature. When most substances are heated, their particles move more vigorously and tend to occupy more space, leading to an increase in dimensions. Conversely, when substances are cooled, they generally contract. This phenomenon occurs in solids, liquids, and gases, although the degree and nature of expansion vary depending on the material's state and properties.

4.1 Linear Expansion

This occurs along a specific dimension or direction, primarily in long, narrow objects (like rods or beams). When the temperature of a solid object increases, its length expands by an amount proportional to its original length and the temperature change. The equation for linear expansion is:

$$\Delta L = \alpha L_0 \Delta T \tag{4.1}$$

where:

- ΔL is the change in length,
- α is the coefficient of linear expansion (unique to each material),
- L_0 is the original length, and
- ΔT is the temperature change.

4.2 Superficial Expansion

Applicable to two-dimensional surfaces, such as sheets or plates. Here, both length and width expand, leading to an increase in surface area. The formula for area expansion is:

$$\Delta A = 2\alpha A_0 \Delta T \tag{4.2}$$

where:

- ΔA is the change in area,
- A_0 is the initial area, and
- ΔT is the temperature change.

4.3 Volumetric Expansion

Relevant for three-dimensional objects (like solids, liquids, and gases). The volume of an object expands with temperature, especially in fluids where this effect is more pronounced. The formula is:

$$\Delta V = \beta V_0 \Delta T \tag{4.3}$$

where:

- ΔV is the change in volume,
- β is the coefficient of volumetric expansion, which is approximately three times the linear expansion coefficient for isotropic solids,
- V_0 is the initial volume, and
- ΔT is the temperature change.

4.4 Thermal Expansion of a Steel Pipeline

A steel section of pipeline is 75 m long when out of service at an ambient temperature of 20°C. In service, it transports steam at 203°C. Assuming the pipe is free to expand, find its length at the operating temperature.
(The coefficient of linear expansion for steel is $\alpha = 12 \times 10^{-6}$ /°C.)

Given:

$$\begin{aligned} L_0 &= 75 \text{ m}, \\ T_1 &= 20^\circ\text{C}, \\ T_2 &= 203^\circ\text{C}, \\ \alpha &= 12 \times 10^{-6} \ /^\circ\text{C}. \end{aligned}$$

The temperature rise is:

$$\Delta T = T_2 - T_1 = 203 - 20 = 183^\circ\text{C}.$$

For linear expansion:

$$L = L_0(1 + \alpha \Delta T).$$

Substitute the values:

$$\begin{aligned} L &= 75\left(1 + 12 \times 10^{-6} \times 183\right) \\ &= 75\left(1 + 0.002196\right) \\ &= 75.1647 \text{ m}. \end{aligned}$$

Final Answer

$$\boxed{L = 75.165 \text{ m}}$$

Note: The pipe expands by:

$$\Delta L = L - L_0 = 0.165 \text{ m} = 165 \text{ mm}.$$

4.4.1 Code

```
# Linear Expansion of a Steel Pipeline
# ------------------------------------
# Given:
#   L0 = 75 m   (initial length)
#   T1 = 20 °C  (ambient temperature)
#   T2 = 203 °C (operating temperature)
#      = 12 × 10  /°C (coefficient of linear expansion for steel)
#
# Find:
#   Final length L at 203 °C.

# Given values
L0 = 75.0                 # m
T1 = 20.0                 # °C
T2 = 203.0                # °C
alpha = 12e-6             # /°C

# Temperature change
delta_T = T2 - T1

# Final length using linear expansion formula
L = L0 * (1 + alpha * delta_T)
```

```
# Expansion amount
expansion = L - L0

# Display results
print(f"Initial Length (m): {L0:.2f}")
print(f"Final Length (m):   {L:.3f}")
print(f"Expansion (m):      {expansion:.3f}")
```

4.5 Thermal Expansion of a Brass Cube

If a solid brass cube measures **50 mm × 50 mm × 50 mm** at **10°C**, what volume will it occupy when heated to **78°C**?

Coefficient of linear expansion for brass, $\alpha = 18.4 \times 10^{-6}\,/^{\circ}\mathrm{C}$.

Step 1: Given Data

$$\begin{aligned} L_0 &= 50 \text{ mm} = 0.05 \text{ m}, \\ T_1 &= 10^{\circ}\mathrm{C}, \\ T_2 &= 78^{\circ}\mathrm{C}, \\ \Delta T &= T_2 - T_1 = 68^{\circ}\mathrm{C}, \\ \alpha &= 18.4 \times 10^{-6}\,/^{\circ}\mathrm{C}. \end{aligned}$$

Step 2: Coefficient of Volumetric Expansion

$$\beta = 3\alpha = 3(18.4 \times 10^{-6}) = 55.2 \times 10^{-6}\,/^{\circ}\mathrm{C}.$$

Step 3: Initial and Final Volumes

Initial volume:

$$V_0 = L_0^3 = (0.05)^3 = 0.000125 \text{ m}^3.$$

Expanded volume:

$$V = V_0(1 + \beta\Delta T)$$

$$V = 0.000125[1 + (55.2 \times 10^{-6})(68)].$$

$$V = 0.000125(1 + 0.0037536) = 0.00012547 \text{ m}^3.$$

Step 4: Final Answer

$$\boxed{V = 0.00012547 \text{ m}^3}$$

Note: The multiplier from m³ to mm³ is 1 × 10 . The brass cube expands from **125000 mm³** to **125469.20 mm³** when heated from **10°C** to **78°C**.

4.5.1 Code

```
# Thermal Expansion of a Brass Cube
# --------------------------------
# Given:
#   L0 = 50 mm  (initial side length)
#   T1 = 10 °C  (initial temperature)
#   T2 = 78 °C  (final temperature)
#     = 18.4 × 10  /°C  (linear expansion coefficient for brass)
#
# Find:
#   Final volume V at 78 °C (in mm³).

# Given values
L0 = 50e-3                # convert mm to m
T1 = 10
T2 = 78
alpha = 18.4e-6           # /°C

# Temperature change
delta_T = T2 - T1

# Volumetric expansion coefficient
beta = 3 * alpha

# Initial and final volumes (in m³)
V0 = L0 ** 3
V = V0 * (1 + beta * delta_T)

# Convert to mm³ for output (1 m³ = 1e9 mm³)
V0_mm3 = V0 * 1e9
V_mm3 = V * 1e9
delta_V = V_mm3 - V0_mm3

# Display results
print(f"Initial Volume (mm³): {V0_mm3:.2f}")
print(f"Final Volume (mm³):   {V_mm3:.2f}")
print(f"Increase in Volume (mm³): {delta_V:.2f}")
```

Chapter 5

Heat Transfer

5.1 Conduction

Conduction transfers heat through **direct contact**, as faster-vibrating molecules pass energy to slower ones. In solids, heat moves from molecule to molecule, and between objects in contact, it flows from the hotter region to the cooler one until thermal equilibrium is reached.

An example of conduction is an iron bar with one end placed in a flame. The other end soon becomes hot as heat is conducted along the bar from molecule to molecule through the metal.

The rate of heat transfer by conduction through a solid is given by:

$$Q = \frac{k\,A\,t\,\Delta T}{s} \tag{5.1}$$

where:

- Q = heat transferred (J)
- k = thermal conductivity of the material (W/m · K)
- A = cross-sectional area through which heat flows (m^2)
- t = time of heat transfer (s)
- ΔT = temperature difference across the material (K or °C)
- s = thickness or length of the material through which heat is conducted (m)

This equation shows that heat conduction increases with greater thermal conductivity, larger surface area, longer time, and larger temperature difference but decreases as the material's thickness increases.

5.2 Temperature Difference Across a Heat Exchanger Endplate

The shell diameter of a large heat exchanger is 1.8 m. The flat endplate (cover) at one end is made of carbon steel, 45 mm thick, and is uninsulated. If the maximum allowable heat loss through the cover, to avoid insulation is 150 MJ/h, determine the temperature difference permitted across the endplate. (For steel, k = 55 W/m°C)

Given

- Shell diameter: (D = 1.8 m)
- Endplate thickness: (s = 45 mm = 0.045 m)
- Material: carbon steel, thermal conductivity: (k = 55 W/m · K)
- Maximum allowable heat loss: ($Q_{\max}$ = 150 MJ/h)
- Endplate is **uninsulated**

Calculate the **temperature difference** ΔT across the endplate to limit heat loss.

1. **Area of the circular endplate**

$$A = \pi \left(\frac{D}{2}\right)^2 = \pi \left(\frac{1.8}{2}\right)^2$$

$$A = \pi \times 0.9^2 = 2.5447 \text{ m}^2$$

2. **Convert heat loss to Watts**

$$Q_{\max} = 150 \text{ MJ/h} = 150 \times 10^6 \text{ J/h}$$

$$1 \text{ h} = 3600 \text{ s} \quad \Rightarrow \quad Q = \frac{150 \times 10^6}{3600} = 41666.7 \text{ W}$$

3. **Use law of conduction for a flat plate**

$$Q = \frac{kA\Delta T}{s} \quad \Rightarrow \quad \Delta T = \frac{Qs}{kA}$$

Substitute values:

$$\Delta T = \frac{41666.7 \times 0.045}{55 \times 2.5447}$$

$$\Delta T = \frac{1875.0}{139.96} = 13.39 \text{ K}$$

Result

$$\boxed{\Delta T = 13.4°\text{C}}$$

Notes:

- The maximum allowable **temperature difference across the endplate** is 13.4°C to keep heat loss below 150 MJ/h.
- If the temperature difference exceeds this, insulation would be required.

5.2.1 Code

```python
# Given data
D = 1.8                 # diameter of endplate, m
L = 0.045               # thickness, m
k = 55                  # thermal conductivity, W/m.K
Q_MJ_per_h = 150        # maximum heat loss, MJ/h

# Convert heat loss to W
Q_W = Q_MJ_per_h * 1e6 / 3600  # W

# Area of the circular plate
import math
A = math.pi * (D/2)**2

# Temperature difference
delta_T = Q_W * L / (k * A)

# Display results
print(f"Endplate area: {A:.4f} m²")
print(f"Heat loss: {Q_W:.2f} W")
print(f"Temperature difference across the endplate: {delta_T:.2f} °C")
```

5.3 Convection Heat Transfer

Convection transfers heat through the movement of fluids (liquids or gases). When a fluid is heated, it expands, becomes less dense, and rises, while cooler, denser fluid moves in to replace it, creating a convection current that distributes heat.

Natural convection occurs without mechanical aid, whereas forced convection involves devices such as pumps or fans. When fluid movement is driven by a pump or fan, heat is transferred by **forced convection**. Examples include:

- A pump circulating hot water through a building's heating system,
- A fan forcing air through an automobile radiator, or
- A forced draft fan pushing hot gases through a boiler.

The **total heat transferred** between a solid surface and a moving fluid over a specified time is given by:

$$Q = h_A \, A \, t \, (T_s - T_f) \tag{5.2}$$

where:

- Q = total heat transferred (J)
- h_A = surface (convective) heat transfer coefficient ($\mathrm{W/m^2 \cdot K}$)
- A = surface area of heat transfer ($\mathrm{m^2}$)
- t = time during which heat transfer occurs (s), with $t = 1\,\mathrm{s}$ for unit time
- T_s = surface temperature of the solid (K or °C)
- T_f = temperature of the surrounding fluid (K or °C)

This equation describes how heat energy is transferred between a surface and a fluid over a unit time of 1 second, depending on the **temperature difference**, the **surface area**, the **time**, and the fluid's heat-carrying effectiveness, represented by the **convective heat transfer coefficient** (h_A)

5.4 Hot Metal Convection Heat Transfer

A hot metal plate measuring **1.2 m × 0.8 m** is exposed to air at **25°C**. The surface temperature of the plate is maintained at **85°C**. If the **convective heat transfer coefficient** (surface heat transfer coefficient) between the plate and air is $h_A = 18\,\mathrm{W/m^2 \cdot {}^\circ C}$, calculate the **rate of heat loss by convection** from the entire plate surface.

Given:

- Plate dimensions: (1.2 m × 0.8 m)
- Plate surface area: $A = 1.2 \times 0.8 = 0.96\,\mathrm{m^2}$
- Surface temperature: $T_s = 85^\circ\mathrm{C}$
- Air temperature: $T_f = 25^\circ\mathrm{C}$
- Convection coefficient: $h_A = 18\,\mathrm{W/m^2 \cdot {}^\circ C}$

Solution

The rate of convective heat loss is given by:

$$Q = h_A \, A \, (T_s - T_f)$$

Substituting the values:

$$Q = 18 \times 0.96 \times (85 - 25)$$

$$Q = 18 \times 0.96 \times 60$$

$$Q = 1036.8 \, \mathrm{W}$$

Answer

$$\boxed{Q = 1036.8 \, \mathrm{W}}$$

5.4.1 Code

```
# Given data
L = 1.2         # length of the plate (m)
W = 0.8         # width of the plate (m)
T_surface = 85 # surface temperature (°C)
T_air = 25      # air temperature (°C)
hA = 18         # convective heat transfer coefficient (W/m²·°C)

# Calculate area of the plate
A = L * W

# Calculate heat loss rate
Q = hA * A * (T_surface - T_air)

# Display results
print(f"Plate area: {A:.2f} m²")
print(f"Temperature difference: {T_surface - T_air:.1f} °C")
print(f"Convective heat loss: {Q:.2f} W")
```

5.5 Radiation

Radiation transfers heat through electromagnetic waves that travel in straight lines and can pass through a vacuum. When these waves strike a surface, they may be absorbed (increasing temperature), reflected, or transmitted. Dark, rough surfaces absorb more radiation, while shiny, smooth ones reflect it.

The heat transfer by radiation is given by:

$$Q = \sigma \, \varepsilon \, A \, t \, (T_1^4 - T_2^4) \tag{5.3}$$

where:

- Q is the **heat energy transferred by radiation** (in kilojoules, kJ).
- σ is the **Stefan–Boltzmann constant**, equal to 5.6703×10^{-11} kW $\cdot$ m^{-2} $\cdot$ K^{-4}.
- ε is the **emissivity** of the surface (dimensionless, between 0 and 1), indicating how efficiently the surface emits or absorbs radiation compared to a perfect blackbody.
- A is the **surface area** of the radiating body (in square meters, m^2).
- t is the **time** during which heat transfer occurs (in seconds, s).
- T_1 is the **absolute temperature of the radiating surface** (in kelvin, K).
- T_2 is the **absolute temperature of the surroundings** (in kelvin, K).

This equation expresses the **net radiant heat transfer** between two bodies at different temperatures, taking into account the emissivity of the surface, its area, and the duration of heat exchange.

Examples include heat from the sun reaching Earth and radiant heat in a boiler furnace.

In a steam boiler, radiation occurs in the furnace. Any heating surfaces that are directly exposed to the furnace will receive heat directly by radiation from the flame. These include the waterwalls and some generating tubes of a watertube boiler, radiant superheater tubes (located at the outlet of the furnace), and the furnace walls of a firetube boiler.

5.6 Radiant Heat from a Flat Circular Plate

A flat circular plate is **400 mm** in diameter. Calculate the theoretical quantity of heat radiated per hour when its temperature is **227 °C** and the temperature of its surrounds is **27 °C**.

Given:

- Stefan–Boltzmann constant: $\sigma = 5.6703 \times 10^{-11}$ kW m^{-2} K^{-4}
- Emissivity: $\varepsilon = 1$ (ideal black body)

1. **Convert temperatures to Kelvin**

$$T_1 = 227 + 273 = 500 \text{ K}, \quad T_2 = 27 + 273 = 300 \text{ K}$$

2. **Calculate the area of the circular plate**

$$A = \pi \left(\frac{D}{2}\right)^2 = \pi \left(\frac{0.4}{2}\right)^2 \approx 0.125664 \text{ m}^2$$

3. **Apply the formula**

$$Q = \sigma \times \varepsilon \times A \times t \times (T_1^4 - T_2^4)$$

Substituting the values:

$$Q = 5.6703 \times 10^{-11} \times 1 \times 0.125664 \times 3600 \times (500^4 - 300^4)$$
$$Q = 1395.46 \text{ kWh}$$

Result

$$\text{Plate area: } 0.125664 \text{ m}^2$$
$$\text{Emissivity: } \varepsilon = 1$$
$$\boxed{\text{Heat radiated (per hour): } 1395.46 \text{ kWh}}$$

Note: This represents the **theoretical maximum radiation** for a perfect black body. Real materials would radiate less depending on their emissivity $\varepsilon < 1$.

5.6.1 Code

```
import math

# Given data
sigma = 5.6703e-11   # kW/m^2/K^4
epsilon = 1.0       # emissivity (black body)
T1 = 227 + 273   # K (plate temperature)
T2 = 27 + 273    # K (surroundings)
d = 400 / 1000     # m (diameter)
A = math.pi * (d / 2)**2  # area in m²
```

```
# Heat radiated per second (kW)
Q_dot = sigma * epsilon * A * (T1**4 - T2**4)

# Heat radiated per hour (kWh)
Q_hour = Q_dot * 3600

# Display results
print(f"Plate area: {A:.6f} m²")
print(f"Emissivity (theoretical): {epsilon}")
print(f"Heat radiated per second: {Q_dot:.4f} kW")
print(f"Heat radiated per hour: {Q_hour:.2f} kWh")
```

Appendix A

SI System Common Mistakes

Using the SI system correctly is crucial for clear communication in science and engineering. Below are common mistakes in using the SI system, examples of incorrect usage, and how to correct them.

Table A.1: SI system rules and common mistakes

Concept	Mistake	Correct Usage	Notes
Use of SI Unit Symbols	`m./s`	`m/s`	Use the correct format without additional punctuation.
Spacing Between Value & Unit	`10kg`	`10 kg`	Always leave a space between the number and the unit symbol.
Incorrect Unit Symbols	`sec`, `hrs`, `°K`	`s`, `h`, `K`	Use the proper SI symbols; symbols are case-sensitive.
Abbreviations for Units	`5 kilograms (kgs)`	`5 kilograms (kg)`	Avoid informal abbreviations like "kgs"; adhere to standard symbols.

Concept	Mistake	Correct Usage	Notes
Multiple Units in Expressions	5 m/s/s, 5 kg/meter^2	5 m/s^2, 5 kg/m^2	Use compact, standardized formats for derived units.
Incorrect Use of Prefixes	0.0001 km	100 mm	Choose prefixes to keep numbers in the range (0.1 x < 1000).
Misplaced Unit Symbols	5/s, kg10	5 s^1, 10 kg	Symbols must follow numerical values, not precede them.
Degrees Celsius vs. Kelvin	300°K	300 K	Kelvin is written without "degree"
Singular vs. Plural Units	5 kgs, 1 meters	5 kg, 1 meter	Symbols do not pluralize; full unit names follow grammar rules.
Capitalization of Symbols	Kg, S, Km, MA	kg, s, km, mA	Symbols are case-sensitive; use uppercase only where specified (e.g., N, Pa).
Capitalization of Unit Names	Newton, Pascal, Watt	newton, pascal, watt	Unit names are lowercase, even if derived from a person's name, unless starting a sentence.
Prefix Capitalization	MilliMeter, MegaWatt	millimeter, megawatt	Prefixes are lowercase for (10^{-1}) to (10^{-9}), uppercase for (10^6) and larger (except k for kilo).
Formatting in Reports	5, Temperature: 300	5 kg, Temperature: 300 K	Always specify units explicitly.

Appendix B

Greek Letters

The following tables present the names of Greek letters and selected symbols commonly used in engineering courses, ensuring precise reference and avoiding reliance on informal descriptors such as “squiggle.”

Table B.1: Greek letters.

Lower Case	Upper Case	Name
α	A	alpha
β	B	beta
γ	Γ	gamma
δ	Δ	delta
ϵ	E	epsilon
ζ	Z	zeta
η	E	eta
θ	Θ	theta
ι	I	iota
κ	K	kappa
λ	Λ	lambda
μ	M	mu
ν	N	nu
ξ	Ξ	xi
o	O	omicron
π	Π	pi
ρ	P	rho
σ	Σ	sigma
τ	T	tau
υ	Υ	upsilon
ϕ	Φ	phi
χ	X	chi

Lower Case	Upper Case	Name
ψ	Ψ	psi
ω	Ω	omega

Table B.2: Commonly used symbols in engineering courses.

Symbol	Name	Use	Course
Δ	Delta	Change	Thermodynamics
Δ	Delta	Displacement	Naval Architecture
∇	Nabla	Volume	Naval Architecture
Σ	Sigma	Sum	Thermodynamics, Naval Architecture, Applied Mechanics
σ	Sigma	Stress	Thermodynamics, Applied Mechanics
ϵ	Epsilon	Modulus of elasticity	Thermodynamics, Applied Mechanics
η	Eta	Efficiency	Thermodynamics
μ	Mu	Friction	Thermodynamics, Applied Mechanics
ω	Omega	Angular velocity	Thermodynamics, Applied Mechanics
ρ	Rho	Density	Thermodynamics, Naval Architecture
τ	Tau	Torque	Thermodynamics, Applied Mechanics

Index