

Files

**Week 13**

# Why files?

- So far, all our examples obtained their input from the keyboard and displayed their output on the Screen.
- However, in many real-life applications, the input data is so much that it will be inconvenient to expect the user to type it each time the program is run.
  - For example: A program to generate employee pay slip from employee data.
- Similarly, there are many applications where the output will be more useful if it is stored in a file rather than the screen.
  - For example: In the program that generates pay slip, how can we print the pay slips and distribute them to the employees if the output is printed on the screen?

# Types of files

```
graph TD; A[Types of files] --> B[Text files]; A --> C[Binary files];
```

## Text files

Used to store data as text.

## Binary files

Used to store data as binary.

# Steps For Using Data Files

1. Declare variables of type **FILE** to represent the files
2. Open the files for reading / writing / appending.
3. Read/write from/to the files.
4. Close the files after processing the data.

# 1. Declare variables of type `FILE`

You can declare the file pointer (binary or text) as the following:

```
FILE *f1; → f1 is a pointer
```

# Open the files for reading/writing

	<b>Text Files</b>	<b>Binary Files</b>
<b>Read</b>	<code>f1 = fopen("data.txt", "r");</code>	<code>f1 = fopen("data.t", "rb");</code>
<b>Write</b>	<code>f1 = fopen("data.txt", "w");</code>	<code>f1 = fopen("data.t", "wb");</code>
<b>append</b>	<code>f1 = fopen("data.txt", "a");</code>	<code>f1 = fopen("data.t", "ab");</code>

# Read/ Write from a file

	<b>Text Files</b>	<b>Binary Files</b>
Read	<code>fscanf</code>	<code>fread</code>
Write	<code>fprintf</code>	<code>fwrite</code>

# Example 1

```
void main(void)
{
    FILE *f1;
    float x=5.346;
    char cc[] = "Hello";

    f1 = fopen("c:\\abc.txt", "wt");
    fprintf( f1, "%s %f", cc, x);
    fclose(f1);
}
```



# Example 2

```
void main(void)
{
    FILE *f1;
    float x;
    char cc[100];

    f1 = fopen("c:\\abc.txt", "rt");
    fscanf( f1, "%s %f", cc, &x);
    fclose(f1);
}
```

# Example 3

```
void main(void)
{
    FILE *f1, *f2;
    char c;

    f1 = fopen("c:\\abc.txt", "rt");
    f2 = fopen("c:\\abc_copy.txt", "wt");
    while(!feof(f1))
        {
            fscanf( f1, "%c", &c);
            fprintf( f2, "%c", c);
        }
    fclose(f1);
    fclose(f2);
}
```

# Example 4

```
struct Pers_Data
{
    char Name[100];
    int ID;
    char Add[255];
};

void main(void)
{
    Pers_Data P[10];
    FILE *f;
    int i;

    for( i = 0; i<10; i++)
    {
        scanf("%s", P[i].Name);
        scanf("%d", &P[i].ID);
        scanf("%s", P[i].Add);
    }

    f = fopen("c:\\DataBase.1", "wt");

    for( i = 0; i<10; i++)
    {
        fprintf(f, "%s", P[i].Name);
        fprintf(f, "%d", P[i].ID);
        fprintf(f, "%s", P[i].Add);
    }
}
//main
```

# Example 5

```
void main(void)
{
    FILE *f1;
    float x=5.346;
    char cc[] = "Hello";

    f1 = fopen("c:\\abc.bin", "wb");
    fwrite(&x, sizeof(float), 1, f1);
    fwrite(cc, sizeof(char), strlen(cc), f1);
    fclose(f1);
}
```

# Example 6

```
void main(void)
{
    FILE *f1;
    float x;
    char cc[100];

    f1 = fopen("c:\\abc.bin", "rb");
    fread(&x, sizeof(float), 1, f1);
    fread(cc, sizeof(char), 5, f1);
    fclose(f1);
}
```

# Example 7

```
void main(void)
{
    FILE *f1, *f2;
    char c;

    f1 = fopen("c:\\abc.bin", "rb");
    f2 = fopen("c:\\abc_copy.bin", "wb");
    while(!feof(f1))
        {
            fread(&c, sizeof(char), 1, f1);
            fwrite(&c, sizeof(char), 1, f2);
        }
    fclose(f1);
    fclose(f2);
}
```

# Example 8

```
struct Pers_Data
{
    char Name[100];
    int ID;
    char Add[255];
};

void main(void)
{
    Pers_Data P[10];
    FILE *f;
    int i;

    for( i = 0; i<10; i++)
    {
        scanf("%s", P[i].Name);
        scanf("%d", &P[i].ID);
        scanf("%s", P[i].Add);
    }

    f = fopen("c:\\DataBase.2", "wb");

    for(i = 0; i<10; i++)
        fwrite(&P[i], sizeof(Pers_Data), 1, f);

} //main
```

# Example 9

```
struct Pers_Data
{
    char Name[100];
    int ID;
    char Add[255];
};

void main(void)
{
    Pers_Data P[10];
    FILE *f;
    int i;

    for( i = 0; i<10; i++)
    {
        scanf("%s", P[i].Name);
        scanf("%d", &P[i].ID);
        scanf("%s", P[i].Add);
    }

    f = fopen("c:\\DataBase.3", "wb");
    fwrite(P, sizeof(Pers_Data), 10, f);
} //main
```