C library function - clock()

Description

The C library function **clock_t clock(void)** returns the number of clock ticks elapsed since the program was launched. To get the number of seconds used by the CPU, you will need to divide by CLOCKS_PER_SEC.

On a 32 bit system where CLOCKS_PER_SEC equals 1000000 this function will return the same value approximately every 72 minutes.

Declaration

Following is the declaration for clock() function.

```
clock_t clock(void)
```

Parameters

- **NA**

Return Value

This function returns the number of clock ticks elapsed since the start of the program. On failure, the function returns a value of -1.

Example

The following example shows the usage of clock() function.

Live Demo

```
#include <time.h>
#include <stdio.h>

int main () {
   clock_t start_t, end_t, total_t;
   int i;

   start_t = clock();
   printf("Starting of the program, start_t = %ld\n", start_t);

   printf("Going to scan a big loop, start_t = %ld\n", start_t);
   for(i=0; i< 10000000; i++) {
   }
```

```
   end_t = clock();
   printf("End of the big loop, end_t = %ld\n", end_t);

   total_t = (double)(end_t - start_t) / CLOCKS_PER_SEC;
   printf("Total time taken by CPU: %f\n", total_t  );
   printf("Exiting of the program...\n");

   return(0);
}
```

Let us compile and run the above program that will produce the following result −

```
Starting of the program, start_t = 0
Going to scan a big loop, start_t = 0
End of the big loop, end_t = 20000
Total time taken by CPU: 0.000000
Exiting of the program...
```