

```

1  using System;
2
3  namespace CS
4  {
5      class Program
6      {
7          static void Main(string[] args)
8          {
9              /*
10               This is my first program
11              */
12              Console.WriteLine("Hello World!");
13
14              Console.ReadLine();
15          }
16      }
17  }
18
19
20
21  /* ===== Variables and DataType ===== */
22
23  using System;
24
25  namespace CS
26  {
27      class Program
28      {
29          static void Main(string[] args)
30          {
31              string Name = "Mehedi Hasan";
32              int age = 20;
33
34
35              Console.WriteLine("The person \n name is " + Name);
36              Console.WriteLine("The age is " + age + " Years Old");
37
38              // ***** Different type of data *****
39
40              string phrase = "Mehedi is good boy";
41              char ch = 'A';
42              int num = 3030;
43              double gpa = 3.4;
44              float num2 = 202.3;
45              bool isMale = true;
46              decimal account = 100.2;
47
48              Console.ReadLine();
49          }
50      }
51  }
52
53
54  // ===== Working With Strings =====
55
56  using System;
57
58  namespace CS
59  {
60      class Program
61      {
62          static void Main(string[] args)
63          {
64

```

```

65 string Name = "Mehedi Hasan" + " is a good boy";
66
67 Console.WriteLine("Mehedi \n Hasan");
68 Console.WriteLine(Name);
69 Console.WriteLine("length of String " + Name.Length ); // 26
70
71 Console.WriteLine("Upper case " + Name.ToUpper()); // MEHEDI HASAN IS A GOOD BOY
72 Console.WriteLine("Lower case " + Name.ToLower()); // mehedi hasan is a good boy
73
74 Console.WriteLine("search for " + Name.Contains("is")); // mehedi hasan is a good boy
75 Console.WriteLine("index by = " + Name[0]);
76
77 Console.WriteLine("index position = " + Name.IndexOf("Hasan")); // 7
78 Console.WriteLine("character index = " + Name.IndexOf('M')); // 0
79
80 Console.WriteLine("substring make = " + Name.Substring(0)); // Subtring( startPos) => Mehedi Hasan is a good boy
81 Console.WriteLine("substring make = " + Name.Substring(0, 3)); // Subtring( startPos, howMany) => Meh
82 }
83 }
84 }
85
86
87 // ===== Working with Number =====
88
89 using System;
90
91 namespace CS
92 {
93     class Program
94     {
95         static void Main(string[] args)
96         {
97
98             int num1 = 5;
99             int num2 = 2;
100             float num3 = 2.0F; // F
101             Console.WriteLine(num1 / num2); // 2
102             Console.WriteLine(num1 / num3); // 2.5
103
104             Console.WriteLine(Math.Abs(-40)); // 40
105             Console.WriteLine(Math.Pow(2, 3)); // 8
106             Console.WriteLine(Math.Sqrt(36)); // 6
107             Console.WriteLine(Math.Max(4, 90)); // 90
108             Console.WriteLine(Math.Min(4, 50)); // 4
109             Console.WriteLine(Math.Round(4.6)); // 5
110
111         }
112     }
113 }
114
115
116 // ===== Getting User Input =====
117
118 using System;
119
120 namespace CS
121 {
122     class Program
123     {
124         static void Main(string[] args)
125         {
126
127             string Name, Age;
128             Console.Write("Enter you Name ");
129             Name = Console.ReadLine(); // Always String return the ReadLine();
130             Console.WriteLine("Enter your Age ");

```

```

130 Console.WriteLine( "Enter your Age ",
131 Age = Console.ReadLine());
132 Console.WriteLine("Hi " + Name + " you are " + Age);
133
134 // how to make number From string
135 // if we add to string number that will only concatenate just
136 Console.WriteLine("44" + "56"); // 4456
137
138 int num = Convert.ToInt32("44");
139 Console.WriteLine(num + 6); // 40
140
141 // input number and convert
142 int num2 = Convert.ToInt32(Console.ReadLine());
143
144 double num3 = Convert.ToDouble(Console.ReadLine());
145
146 Console.ReadLine();
147
148 }
149 }
150 }
151
152
153 // ===== Array =====
154 using System;
155
156 namespace CS
157 {
158     class Program
159     {
160         static void Main(string[] args)
161         {
162             int[] Arr = {1, 2, 3, 4};
163             Console.WriteLine(Arr[0]);
164
165             // if we do not want to give the initially value then this way we have to follow
166             string[] Names = new string[300]; // 300 is size we have to tell the
167
168             Names[0] = "mehedi";
169             Names[1] = "SM Nayema";
170
171         }
172     }
173 }
174
175
176 // ===== Methods or function =====
177 using System;
178
179 namespace CS
180 {
181     class Program
182     {
183         // Main() function
184         static void Main(string[] args)
185         {
186
187             // Call ours Method...
188             SayHi("Mehedi", 32);
189
190             // Add function Call
191             int x = add(2, 3);
192
193             Console.WriteLine(x); // 5
194         }
195

```

```

196 // New Method which we want to create ...
197 static void SayHi(string name, int age)
198 {
199     Console.WriteLine("Hi " + name + " age is " + age);
200 }
201
202 static int add(int a, int b){
203
204     return (a + b);
205
206 }
207
208 }
209 }
210
211
212 // ===== if else =====
213 using System;
214
215 namespace CS
216 {
217     class Program
218     {
219         // Main() function
220         static void Main(string[] args)
221         {
222             bool isMale = true;
223             bool isTall = true;
224
225             if(isMale == true && isTall == true)
226             {
227                 Console.WriteLine("you are Male and Tall");
228             }
229             else if(isMale == false)
230             {
231                 Console.WriteLine("you are not Male or small");
232             }
233             else
234             {
235                 Console.WriteLine("Other Gender");
236             }
237         }
238     }
239 }
240
241
242 // ===== loop =====
243 using System;
244
245 namespace CS
246 {
247     class Program
248     {
249         // Main() function
250         static void Main(string[] args)
251         {
252             int i = 1;
253
254             while(i <= 10)
255             {
256                 Console.WriteLine(i);
257                 i++;
258             }
259
260             int[] Arr = {1, 2, 3, 4, 5, 6};

```

```

261         for(int j = 0; j < 5; j++)
262         {
263             Console.Write(Arr[j] + " ");
264         }
265     }
266 }
267 }
268 }
269
270
271 // ===== 2D Array =====
272
273 using System;
274
275 namespace CS
276 {
277     class Program
278     {
279         // Main() function
280         static void Main(string[] args)
281         {
282             int [, ] numberGrid = {
283
284                 {1, 2, 3},
285                 {4, 5, 6},
286                 {7, 8, 9}
287
288             };
289
290             // [0, 1] => 2 here
291             Console.WriteLine(numberGrid[0, 1]); // name[row , col];
292
293             // But if we do not mention element of THEN we have to tell the size...
294
295             int [, ] Arr2 = new int [2, 3]; // row = 2, col = 3
296
297
298         }
299     }
300 }
301
302 // ===== Exception Handling =====
303 using System;
304
305 namespace CS
306 {
307     class Program
308     {
309         // Main() function
310         static void Main(string[] args)
311         {
312
313             try
314             {
315                 Console.WriteLine("Enter a number ");
316                 int num1 = Convert.ToInt32(Console.ReadLine());
317                 Console.WriteLine("Enter another Number ");
318                 int num2 = Convert.ToInt32(Console.ReadLine());
319
320                 Console.WriteLine( num1 / num2);
321             }
322
323             catch( DivideByZeroException e)
324             {
325
326                 Console.WriteLine("divided by Zero");

```

```

326         Console.WriteLine( divided by zero ),
327     }
328     catch(FormatException e)
329     {
330         Console.WriteLine("Format Exception")
331     }
332
333     catch(Exception e)
334     {
335
336         Console.WriteLine("Error" + e.Message);
337     }
338     finally{
339         Console.WriteLine("Run all time")
340     }
341
342     Console.ReadLine();
343 }
344 }
345 }
346
347
348 // ===== Classes & Objects =====
349
350 // ***** main() function *****
351 using System;
352
353 namespace CS
354 {
355     class Program
356     {
357         // Main() function
358         static void Main(string[] args)
359         {
360             // this how we can create object of Book class ..
361             /*
362              dataType variable;
363              string bookName = "C programming";
364              Book variable = create object
365
366              Access Atributes by (.) dot of the class
367
368              */
369             Book book1 = new Book();
370
371             book1.Title = "Harry Pottter";
372             book1.author = "Jk Rowling";
373             book1.pages = 400;
374
375             Console.WriteLine(book1.Title);
376         }
377     }
378 }
379
380 // ***** END *****
381
382 // ***** Book Class *****
383
384 using System;
385
386 namespace CS
387 {
388     class Book
389     {
390         public string Title;
391         public string author;

```

```

392     public int pages;
393 }
394
395 }
396
397 // ***** END *****
398
399 // ===== Class End =====
400
401
402
403
404
405
406 // ===== Constructors =====
407
408 // ***** main() function *****
409 using System;
410
411 namespace CS
412 {
413     class Program
414     {
415         // Main() function
416         static void Main(string[] args)
417         {
418
419             Book book1 = new Book();
420             Book book2 = new Book("C programming");
421
422             book1.Title = "Harry Pottter";
423             book1.author = "Jk Rowling";
424             book1.pages = 400;
425
426             Console.WriteLine(book1.Title);
427
428             // We can pass all Argument to initializion the Book Atribute
429             // create constructor same way
430             Book book3 = new Book("C++", "Sir", 300);
431
432             Console.WriteLine(book3.Title); // C++
433         }
434     }
435 }
436
437 // **** OUTPUT ****
438 /*
439     Creating book object
440     book name is C programming object create call
441     Harry Pottter
442     C++
443 */
444
445 // ***** END *****
446
447
448 // ***** Book parameters Base constructor *****
449 using System;
450
451 namespace CS
452 {
453     class Book
454     {
455         public string Title;
456         public string author;

```

```

457     public int pages;
458
459     // Constructor ... when object create in main function that time it run
460
461     // Constructor without parameter
462     public Book()
463     {
464         Console.WriteLine("Creating book object");
465     }
466
467     // Constructor ...with parameter
468     public Book(string name){
469         Console.WriteLine("book name is " + name + " object create call");
470     }
471
472     public Book(string aTitle, string aAuthor, int aPages)
473     {
474         Title = aTitle;
475         author = aAuthor;
476         pages = aPages;
477     }
478
479 }
480
481 }
482 // ***** END *****
483
484 // ===== topic End =====
485
486
487
488
489 // ===== Object Methods create and use =====
490
491 // ***** main() function *****
492 using System;
493
494 namespace CS
495 {
496     class Program
497     {
498         // Main() function
499         static void Main(string[] args)
500         {
501             Student St1 = new Student("mehedi", "CSE", 3.4);
502             Student St2 = new Student("Hasan", "CSE", 3.6);
503
504             Console.WriteLine(St1.HasEligable());
505         }
506     }
507 }
508
509 // ***** END *****
510
511
512 // ***** Student Class *****
513 using System;
514
515 namespace CS
516 {
517     class Student
518     {
519         public string Name;
520         public string Subject;
521         public double Gpa;
522     }
523 }

```



```

522
523     public Student(string aName, string aSubject, double aGpa)
524     {
525         Name = aName;
526         Subject = aSubject;
527         Gpa = aGpa;
528     }
529
530     public bool HasEligable()
531     {
532         if(Gpa >= 3.5)
533         {
534             return true;
535         }
536         else
537         {
538             return false;
539         }
540     }
541 }
542 }
543
544 }
545 // ***** END *****
546
547 // ===== topic End =====
548
549
550
551
552
553 // ===== Getter and Setter value as Private =====
554
555 // ***** main() function *****
556
557 using System;
558
559 namespace CS
560 {
561     class Program
562     {
563         // Main() function
564         static void Main(string[] args)
565         {
566             Student St1 = new Student("mehedi", "CSE", 3.4);
567             Student St2 = new Student("Hasan", "CSE", 3.6);
568
569             Console.WriteLine(St1.PPrivateGpa);
570         }
571     }
572 }
573
574
575 // ***** END *****
576
577
578 // ***** Student *****
579 using System;
580
581 namespace CS
582 {
583     class Student
584     {
585         public string Name;
586         public string Subject;
587         // Create variable as private to secure

```

```

587 // Create variable as private to be unique
588 private double PrivateGpa;
589
590 public Student(string aName, string aSubject, double aGpa)
591 {
592     Name = aName;
593     Subject = aSubject;
594     PrivateGpa = aGpa;
595 }
596
597 // Special kind of function which is allow to me getter and setter applying
598 public double PPrivateGpa
599 {
600     get {
601         return PrivateGpa;
602     }
603     set{
604         if(value >= 3.2){
605
606             PrivateGpa = value;
607         }
608         else{
609             PrivateGpa= 0.0;
610         }
611     }
612 }
613
614 }
615
616 }
617 // ***** END *****
618
619 // ===== topic End =====
620
621
622
623
624
625 // ===== topic =====
626
627 // ***** main() function *****
628
629 // ***** END *****
630
631
632 // ***** sub *****
633
634 // ***** END *****
635
636 // ===== topic End =====

```