

```
In [1]: #Name:- Kuldeep Ghorpade  
#Div:- B  
#Roll No. :- 09  
#Experiment No. :-03
```

```
In [2]: #Declaration of Types
```

```
In [3]: (10+20)::Int64
```

```
Out[3]: 30
```

```
In [4]: (10+20)::Float64
```

TypeError: in typeassert, expected Float64, got a value of type Int64

Stacktrace:

```
[1] top-level scope  
  @ In[4]:1  
[2] eval  
  @ .\boot.jl:368 [inlined]  
[3] include_string(mapexpr::typeof(REPL.softscope), mod::Module, code::String, filename::String)  
  @ Base .\loading.jl:1428
```

```
In [5]: local x::Int64=25
```

```
Out[5]: 25
```

```
In [6]: function Addition(x, y)::Float64  
return x + y  
end  
Addition(1, 2)
```

```
Out[6]: 3.0
```

```
In [7]: function Addition(x::Number)  
return x + 1  
end
```

```
Out[7]: Addition (generic function with 2 methods)
```

```
In [8]: Addition(3)
```

```
Out[8]: 4
```

```
In [9]: Addition(7.3)
```

```
Out[9]: 8.3
```

```
In [10]: #Primitive Types
```

```
In [11]: sizeof(Integer)
```

```
Abstract type Integer does not have a definite size.
```

```
Stacktrace:
```

```
[1] sizeof(x::Type)
  @ Base .\essentials.jl:473
[2] top-level scope
  @ In[11]:1
[3] eval
  @ .\boot.jl:368 [inlined]
[4] include_string(mapexpr::typeof(REPL.softscope), mod::Module, code::String, filename::String)
  @ Base .\loading.jl:1428
```

```
In [12]: sizeof(Int32)
```

```
Out[12]: 4
```

```
In [13]: sizeof(Float64)
```

```
Out[13]: 8
```

```
In [14]: #Composite Types
```

```
In [15]: #Object Oriented Programming in Julia
```

```
In [16]: struct MyObj
           field1

           field2
       end
```

```
In [17]: myobj=MyObj("Hello","World")
```

```
Out[17]: MyObj("Hello", "World")
```

```
In [18]: myobj.field1
```

```
Out[18]: "Hello"
```

```
In [19]: myobj.field1="welcome"
```

```
setfield!: immutable struct of type MyObj cannot be changed
```

```
Stacktrace:
```

```
[1] setproperty!(x::MyObj, f::Symbol, v::String)
  @ Base .\Base.jl:39
[2] top-level scope
  @ In[19]:1
[3] eval
  @ .\boot.jl:368 [inlined]
[4] include_string(mapexpr::typeof(REPL.softscope), mod::Module, code::String, filename::String)
  @ Base .\loading.jl:1428
```

```
In [20]: mutable struct Person
         name::String
         age::Float64
         isactive
         end
```

```
In [21]: newperson=Person("Peter",25,true)
```

```
Out[21]: Person("Peter", 25.0, true)
```

```
In [22]: newperson
```

```
Out[22]: Person("Peter", 25.0, true)
```

```
In [23]: newperson.name
```

```
Out[23]: "Peter"
```

```
In [24]: newperson.name="John"
```

```
Out[24]: "John"
```

```
In [25]: newperson
```

```
Out[25]: Person("John", 25.0, true)
```

```
In [26]: mutable struct Person
         name::String
         age::Float64
         isactive
         function Person(name,age=30) #Default Constructor
             new(name,age,true)
         end
         end
```

```
In [27]: newperson=Person("Peter",25)
```

```
Out[27]: Person("Peter", 25.0, true)
```

```
In [28]: function birthday(person::Person)
         person.age+=1
         end
```

```
Out[28]: birthday (generic function with 1 method)
```

```
In [29]: birthday(newperson)
```

```
Out[29]: 26.0
```

```
In [30]: newperson.age
```

```
Out[30]: 26.0
```

In []: