Information Security 3

Sri M J Shankar Raman,

Consultant Department of Computer Science and Engineering,

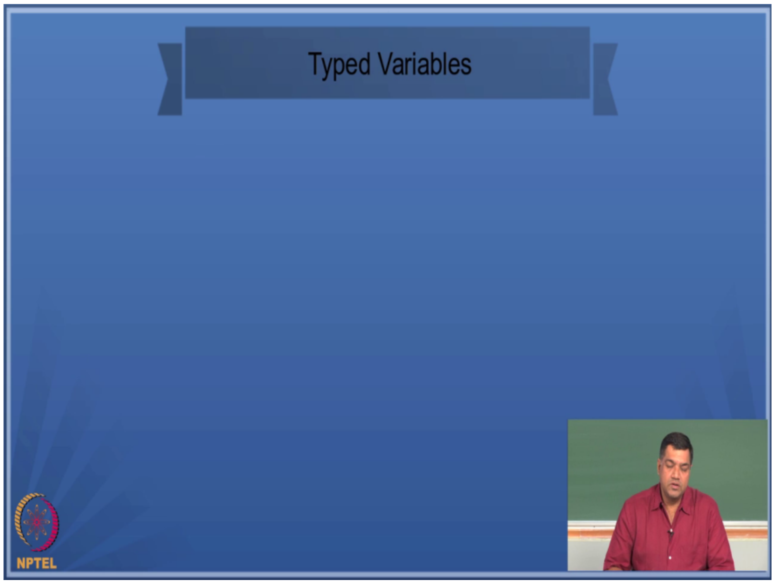
Indian Institute of Technology Madras

Module 42

Shell Declarations

So welcome to this session on shell programming, until now we've been discussing various aspects of shell programming and in the last few classes we've been discussing some advance concepts such as executing the same command twice, ignoring shell commands, ignoring signals, and then handling signals and so on. If you remember during the start of this course, I had informed you that in shell all the variables are considered as string.

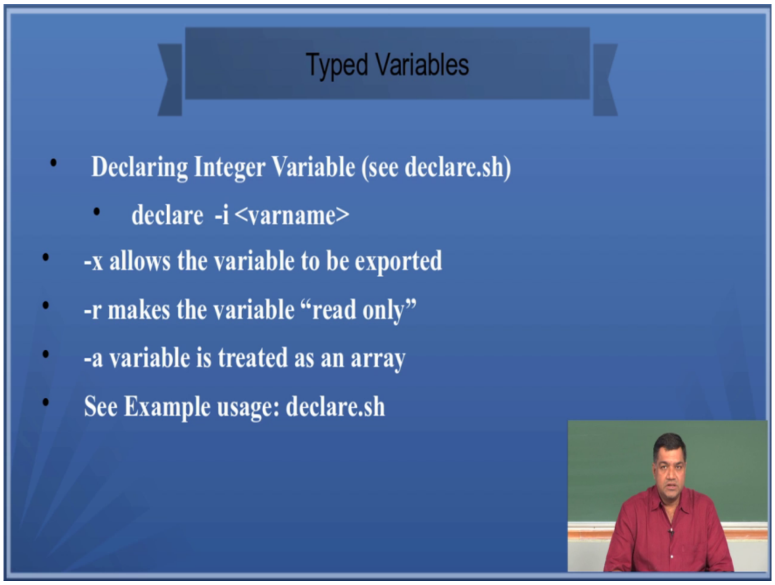
(Refer Slide Time: 00:54)



Typed Variables

Well, we had mentioned that to ensure that you don't have any confusion during the past few classes about how to use and handle variables and as well as being careful while writing the shell programming languages, now what we are going to see in this session is how to declare typed variables, so this will in fact reduce your burden of checking for whether a variable is an integer or a string or whether it's an array etc. So one of the ways to do it is to actually declare what the variable type is and that's exactly what we are going to do with these typed variables.

(Refer Slide Time: 01:42)



Typed Variables

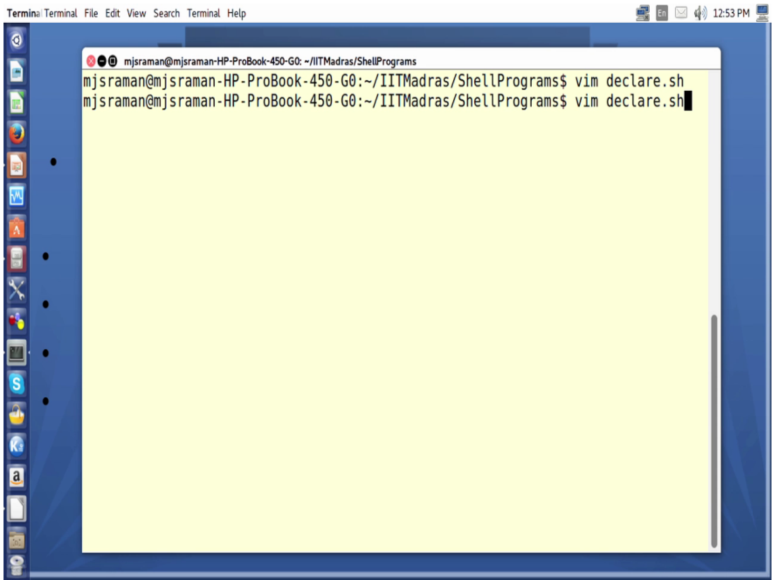
- Declaring Integer Variable (see declare.sh)
  - declare -i <varname>
- -x allows the variable to be exported
- -r makes the variable "read only"
- -a variable is treated as an array
- See Example usage: declare.sh

So the shell provides a way to declare a variable, so for example suppose I am going to use integers ok in my program so what I can do is I can actually declare some variables to be integers by using a command such as declare space minus I space and then whatever variables ok, so the advantage of this is that you can overcome many of the problems associated with converting a variable from a string to an integer and then performing lot of mathematical operations.

So if you remember in one of the classes we had to use either expr or we had to use a calculator and advance calculator such as bc or something to ensure that you are able to do arithmetic operations. Now such problems can be solved if you can actually redefine the variables, so we will see some of the important aspects or the parameters that this declare command takes ok so the, as I discussed previously the minus I tells you that the variable names that is going to follow after this minus I will be integer variables, now minus x allows you to export the variables ok.

So if you remember we had discussed about exporting shell variables previously and you can also make a variable read only, now this is very important because if you want to declare a variable and you don't want any other person to modify that variable then it's nice to make this variable read only, and then we can also declare an array or you can let a variable to be treated as array by using the minus a command, now let us see all of this using example.

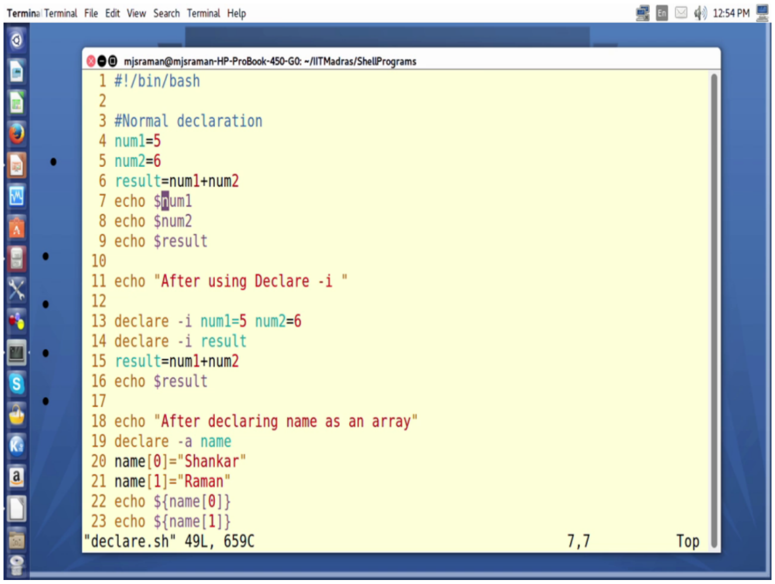
(Refer Slide Time: 03:46)


Terminal Terminal File Edit View Search Terminal Help
12:53 PM
mjsraman@mjsraman-HP-ProBook-450-G0: ~/IITMadras/ShellPrograms

```
mjsraman@mjsraman-HP-ProBook-450-G0:~/IITMadras/ShellPrograms$ vim declare.sh
mjsraman@mjsraman-HP-ProBook-450-G0:~/IITMadras/ShellPrograms$ vim declare.sh
```

So let us try to take this program called declare.sh we'll try to understand the program line by line, we'll also execute it and then we'll see how these program works, so let us first take a look at this declare.sh program.

(Refer Slide Time: 03:58)

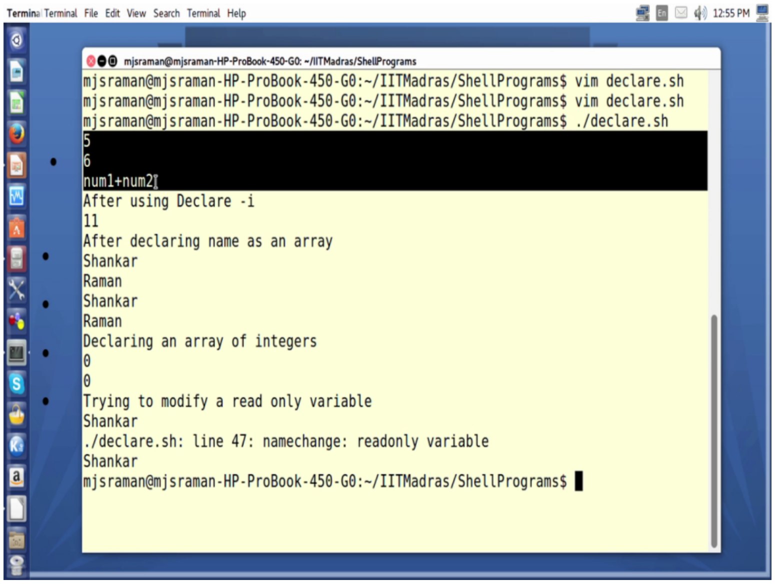

Terminal Terminal File Edit View Search Terminal Help
12:54 PM
mjsraman@mjsraman-HP-ProBook-450-G0: ~/IITMadras/ShellPrograms

```
  1 #!/bin/bash
  2 
  3 #Normal declaration
  4 num1=5
  5 num2=6
  6 result=num1+num2
  7 echo $num1
  8 echo $num2
  9 echo $result
 10 
 11 echo "After using Declare -i "
 12 
 13 declare -i num1=5 num2=6
 14 declare -i result
 15 result=num1+num2
 16 echo $result
 17 
 18 echo "After declaring name as an array"
 19 declare -a name
 20 name[0]="Shankar"
 21 name[1]="Raman"
 22 echo ${name[0]}
 23 echo ${name[1]}
"declare.sh" 49L, 659C                                        7,7           Top
```

So if you remember when we started of our course we said that a variables can be dictate something like this say for example line number 4 say that num1 will take a value of 5 and then num2 will take a value of 6 and I am typing something known as a result which is num1 plus num2 then I am trying to echo num1 num 2 and num3 and if you remember since the shell considered every variable to be a string, if I echo the result here in line number 9.

So what is going to happen is that just say that result is equal to num1 plus num2 but whereas our intention is to add these numbers 5 and 6, so even though it prints num1 to be 5 and num2 to be 6, the result will be printed as 5 plus 6, it doesn't print the value of 11. So let us try to run this program and see whether this what happens ok, so we'll ignore the other values except for the first three echo statements, ok.

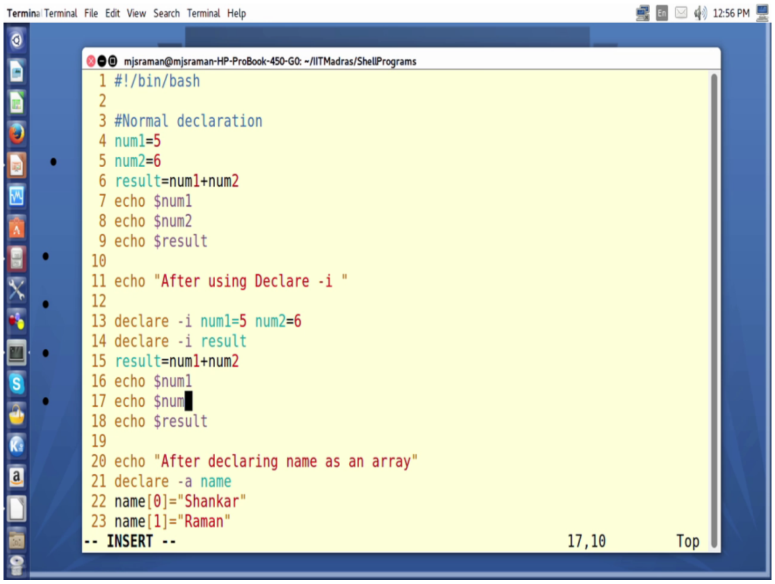
(Refer Slide Time: 04:56)


Terminal Terminal File Edit View Search Terminal Help
12:55 PM
mjsraman@mjsraman-HP-ProBook-450-G0: ~/IITMadras/ShellPrograms

```
mjsraman@mjsraman-HP-ProBook-450-G0:~/IITMadras/ShellPrograms$ vim declare.sh
mjsraman@mjsraman-HP-ProBook-450-G0:~/IITMadras/ShellPrograms$ vim declare.sh
mjsraman@mjsraman-HP-ProBook-450-G0:~/IITMadras/ShellPrograms$ ./declare.sh
5
6
num1+num2
After using Declare -i
11
After declaring name as an array
Shankar
Raman
Shankar
Raman
Declaring an array of integers
0
0
Trying to modify a read only variable
Shankar
./declare.sh: line 47: namechange: readonly variable
Shankar
mjsraman@mjsraman-HP-ProBook-450-G0:~/IITMadras/ShellPrograms$
```

So let me run this program, so declare.sh so let's look at the first three statements which is of importance to us, so it says num1 is 5, num2 is 6 but when I try to print the result what is happening it is printing it as num1 plus num2, Now what I want is number 11, so how do I get this number11.

(Refer Slide Time: 05:18)

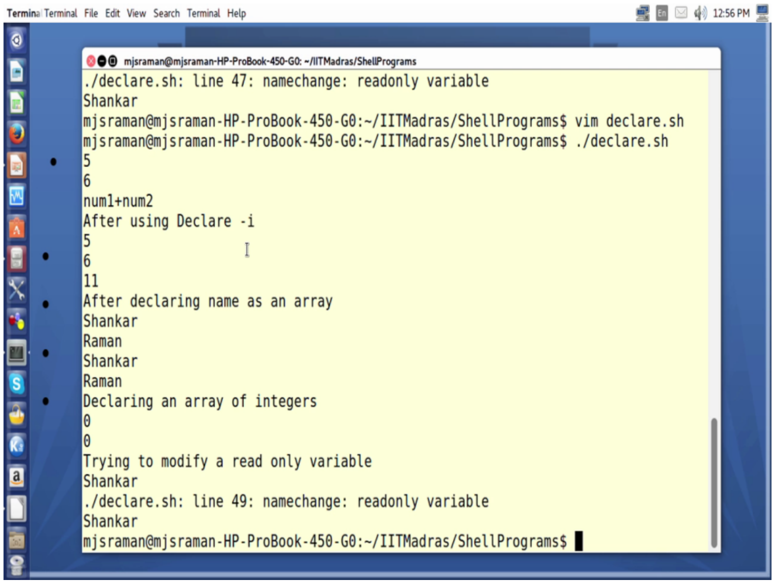

Terminal Terminal File Edit View Search Terminal Help
12:56 PM
mjsraman@mjsraman-HP-ProBook-450-G0: ~/IITMadras/ShellPrograms

```
 1 #!/bin/bash
 2 
 3 #Normal declaration
 4 num1=5
 5 num2=6
 6 result=num1+num2
 7 echo $num1
 8 echo $num2
 9 echo $result
10 
11 echo "After using Declare -i "
12 
13 declare -i num1=5 num2=6
14 declare -i result
15 result=num1+num2
16 echo $num1
17 echo $num
18 echo $result
19 
20 echo "After declaring name as an array"
21 declare -a name
22 name[0]="Shankar"
23 name[1]="Raman"
-- INSERT --                                   17,10          Top
```

So let us now try to go back and look at the next part of the program, so now we are saying in line number 11, we are saying echo after using declare minus I, so here what I am doing in line number 13 and 14 is very important. So I am just saying that I am using this command called declare and then I am saying that whatever is the variables that are going to be here are going to be integers.

Therefore I am, what I am doing is declare space minus I space num1 is equal to 5 and then I am giving num2 is equal to 6, then again in the next line I am declaring the value result. So the result is also I am declaring it as integer and now if you see the best part of it when I do a result is equal to num1 plus num2 and then I try to print the result so what I will do is, I will so print echo dollar num1 and we’ll also print num2, now this will not look anything different from what we’ll printed previously but then the result is what will surprise us.

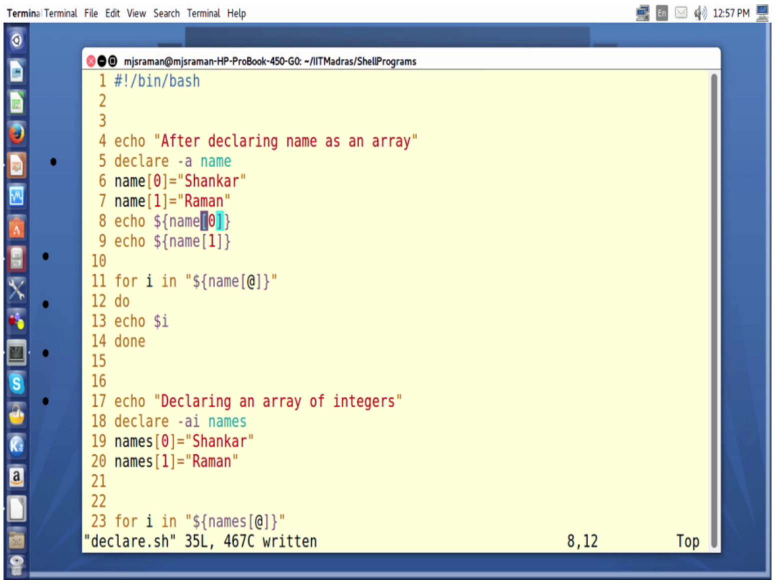
(Refer Slide Time: 06:21)


Terminal Terminal File Edit View Search Terminal Help
12:56 PM
mjsraman@mjsraman-HP-ProBook-450-G0: ~/IITMadras/ShellPrograms

```
./declare.sh: line 47: namechange: readonly variable
Shankar
mjsraman@mjsraman-HP-ProBook-450-G0:~/IITMadras/ShellPrograms$ vim declare.sh
mjsraman@mjsraman-HP-ProBook-450-G0:~/IITMadras/ShellPrograms$ ./declare.sh
5
6
num1+num2
After using Declare -i
5
6
11
After declaring name as an array
Shankar
Raman
Shankar
Raman
Declaring an array of integers
0
0
Trying to modify a read only variable
Shankar
./declare.sh: line 49: namechange: readonly variable
Shankar
mjsraman@mjsraman-HP-ProBook-450-G0:~/IITMadras/ShellPrograms$
```

So if you look at running this program again and then I run this, so if you look at this after declaring minus I, I am able to see the same 5, same 6 but then the output is now, is not num1 plus num2 it is now 11. So we see that if we use the declare statement ok, or the command then these two numbers are considered to be integers, so let us now try to see the other options of declare, so let us go and take the look at this program again, ok?

(Refer Slide Time: 06:56)

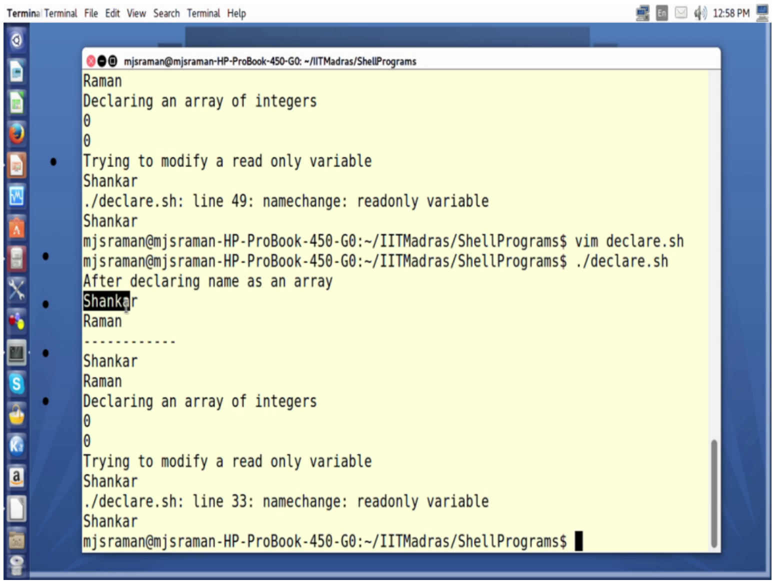

Terminal Terminal File Edit View Search Terminal Help
12:57 PM
mjsraman@mjsraman-HP-ProBook-450-G0: ~/IITMadras/ShellPrograms

```
#!/bin/bash


echo "After declaring name as an array"
declare -a name
name[0]="Shankar"
name[1]="Raman"
echo ${name[0]}
echo ${name[1]}

for i in "${name[@]}"
do
echo $i
done


echo "Declaring an array of integers"
declare -ai names
names[0]="Shankar"
names[1]="Raman"


for i in "${names[@]}"
"declare.sh" 35L, 467C written                              8,12          Top
```

Let us move on we will remove this lines to ensure that it doesn't create any confusion ok. So what we are doing is now, let us now in line number 5 what we are doing is we are declaring named to be an array, so now once you had declared it to be an array, I hope you'll remember how to initialize array values, what we are doing right now is this ok,

So once I declare the name to be an array then I can start giving the indices to the array, so I've given a index of zero, where it given my first name and then the index of 1 where I given my last name and then when I try to print this, so this is how you print the array values if you remember we've to give a dollar symbol and then we've to use this curly braces, so I've printed the value of zero on one, let us see what happens, so here I'll just insert a dummy statement to understand the results much better, ok?

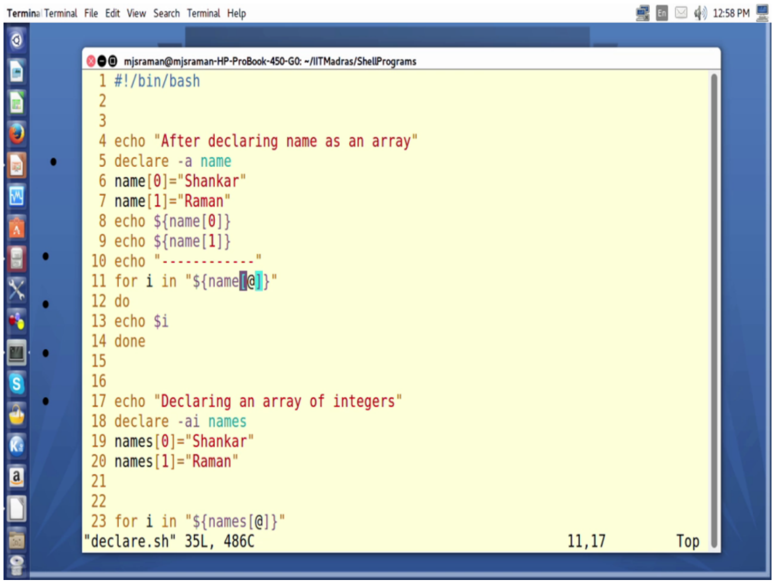
(Refer Slide Time: 07:57)


Termina Terminal File Edit View Search Terminal Help
12:58 PM
mjsraman@mjsraman-HP-ProBook-450-G0: ~/IITMadras/ShellPrograms

```
Raman
Declaring an array of integers
0
0
Trying to modify a read only variable
Shankar
./declare.sh: line 49: namechange: readonly variable
Shankar
mjsraman@mjsraman-HP-ProBook-450-G0:~/IITMadras/ShellPrograms$ vim declare.sh
mjsraman@mjsraman-HP-ProBook-450-G0:~/IITMadras/ShellPrograms$ ./declare.sh
After declaring name as an array
Shankar
Raman
------------
Shankar
Raman
Declaring an array of integers
0
0
Trying to modify a read only variable
Shankar
./declare.sh: line 33: namechange: readonly variable
Shankar
mjsraman@mjsraman-HP-ProBook-450-G0:~/IITMadras/ShellPrograms$ 
```

So what I do is now run this program, So if you run this program if you look at this after declaring the name as an array when I print name of zero and name of one what I get is my name gets printed and so here is what the delay meter that we put, now this value can be printed in another fashion.

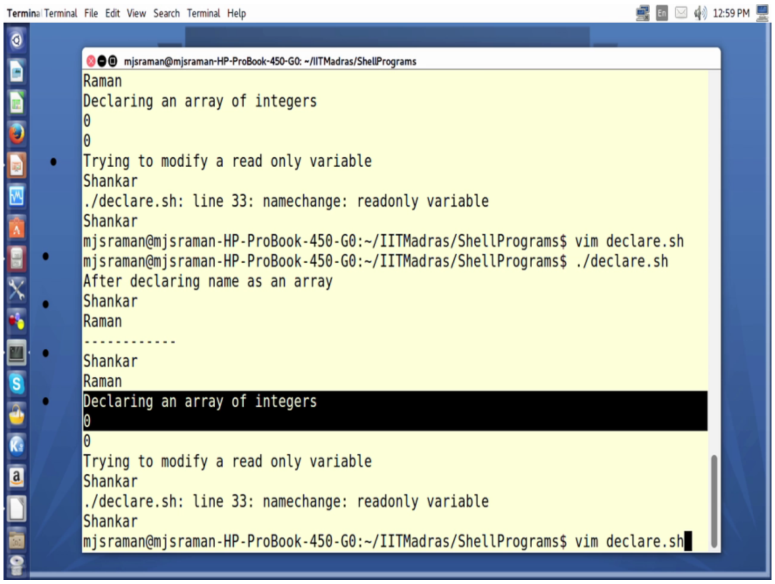
(Refer Slide Time: 08:20)


Terminal Terminal File Edit View Search Terminal Help
12:58 PM
mjsraman@mjsraman-HP-ProBook-450-G0: ~/IITMadras/ShellPrograms

```
#!/bin/bash


echo "After declaring name as an array"
declare -a name
name[0]="Shankar"
name[1]="Raman"
echo ${name[0]}
echo ${name[1]}
echo "-----------"
for i in "${name[@]}"
do
echo $i
done


echo "Declaring an array of integers"
declare -ai names
names[0]="Shankar"
names[1]="Raman"


for i in "${names[@]}"
"declare.sh" 35L, 486C                                  11,17         Top
```

So that is what we’ve done in the next line of the program, so if you look at the next line of the program instead of printing name zero and name one, we’ve just said that now if you remember this expands the array, ok so it expands this to name of zero and the name of one which is nothing but Shankar and Raman, so again if you print this, this line so if you can see the echo before declaring an array of integers.

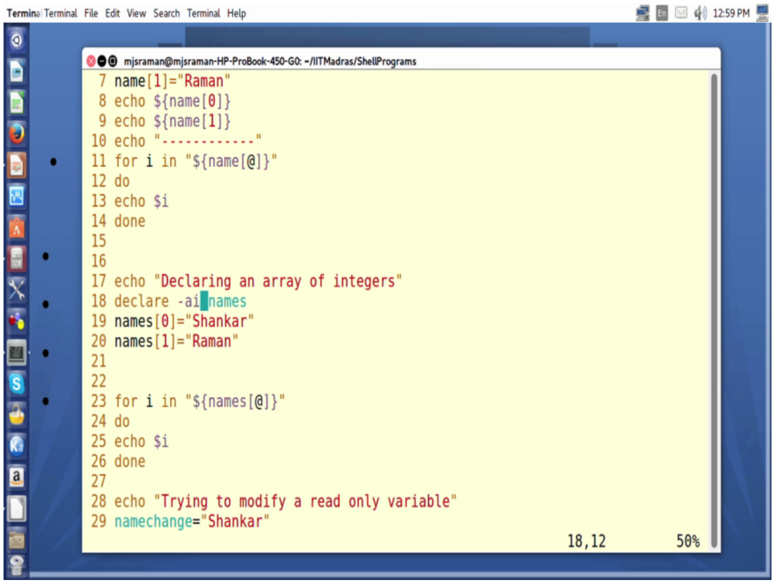
(Refer Slide Time: 08:45)


Terminal Terminal File Edit View Search Terminal Help
12:59 PM
mjsraman@mjsraman-HP-ProBook-450-G0: ~/IITMadras/ShellPrograms

```
Raman
Declaring an array of integers
0
0
Trying to modify a read only variable
Shankar
./declare.sh: line 33: namechange: readonly variable
Shankar
mjsraman@mjsraman-HP-ProBook-450-G0:~/IITMadras/ShellPrograms$ vim declare.sh
mjsraman@mjsraman-HP-ProBook-450-G0:~/IITMadras/ShellPrograms$ ./declare.sh
After declaring name as an array
Shankar
Raman
-----------
Shankar
Raman
Declaring an array of integers
0
0
Trying to modify a read only variable
Shankar
./declare.sh: line 33: namechange: readonly variable
Shankar
mjsraman@mjsraman-HP-ProBook-450-G0:~/IITMadras/ShellPrograms$ vim declare.sh
```

We can see that this prints my name and this is another way of printing, so this is another way of printing the result ok. Now as a next step, let's try to declare an array of integers which essentially means we are going to use both the minus a and the minus i option.

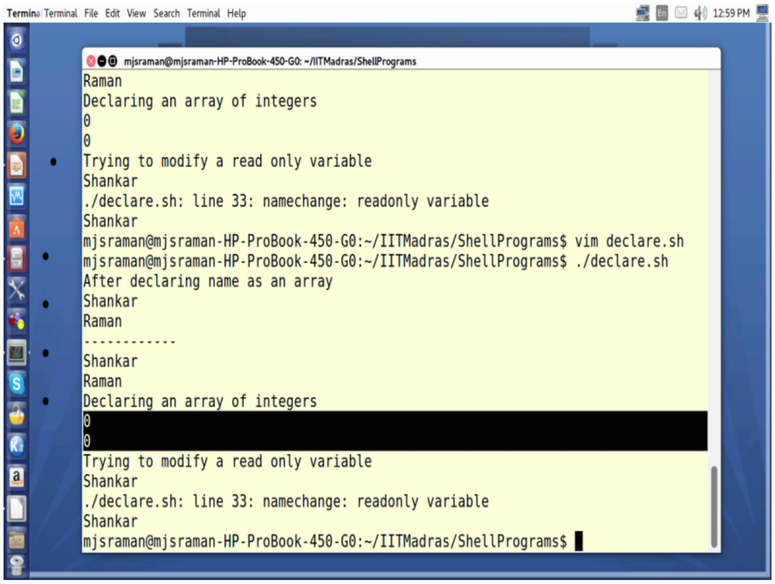
(Refer Slide Time: 09:02)


Terminal Terminal File Edit View Search Terminal Help
12:59 PM
mjsraman@mjsraman-HP-ProBook-450-G0: ~/IITMadras/ShellPrograms

```
name[1]="Raman"
echo ${name[0]}
echo ${name[1]}
echo "-----------"
for i in "${name[@]}"
do
echo $i
done


echo "Declaring an array of integers"
declare -ai names
names[0]="Shankar"
names[1]="Raman"


for i in "${names[@]}"
do
echo $i
done

echo "Trying to modify a read only variable"
namechange="Shankar"
                                                18,12         50%
```

Now if you look at this I am trying to initialize them as integers, but then what I am trying to do is I am trying to print the message string because initially I told you that any variable is actually treated as a string. Now in this case if you see even though I am declaring it as integer and then I am putting the string values so when I try to print it, let's see what happens, ok?

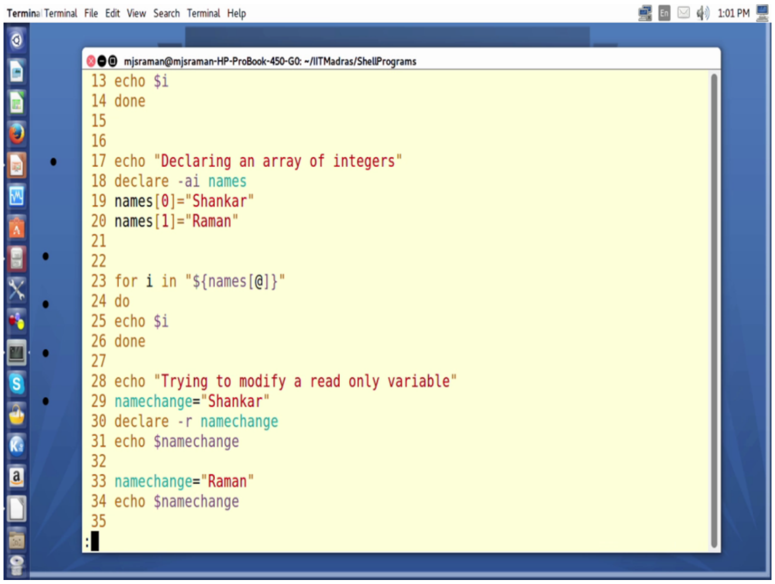
(Refer Slide Time: 09:33)


Terminal Terminal File Edit View Search Terminal Help
12:59 PM
mjsraman@mjsraman-HP-ProBook-450-G0: ~/IITMadras/ShellPrograms

```
Raman
Declaring an array of integers
0
0
Trying to modify a read only variable
Shankar
./declare.sh: line 33: namechange: readonly variable
Shankar
mjsraman@mjsraman-HP-ProBook-450-G0:~/IITMadras/ShellPrograms$ vim declare.sh
mjsraman@mjsraman-HP-ProBook-450-G0:~/IITMadras/ShellPrograms$ ./declare.sh
After declaring name as an array
Shankar
Raman
------------
Shankar
Raman
Declaring an array of integers
0
0
Trying to modify a read only variable
Shankar
./declare.sh: line 33: namechange: readonly variable
Shankar
mjsraman@mjsraman-HP-ProBook-450-G0:~/IITMadras/ShellPrograms$
```

So we'll have to observe what is there before the read only variable, so let's try to print this so if you look at this, I define Shankar and Raman which a strings but then when I print them because you've declared them as integers, it says those values are not there. Therefore it is just printing a integer value of zero which is essentially the value that it got initialize to, now let's look at the other option of making a variable read only so what we are trying to do is,

(Refer Slide Time: 09:57)

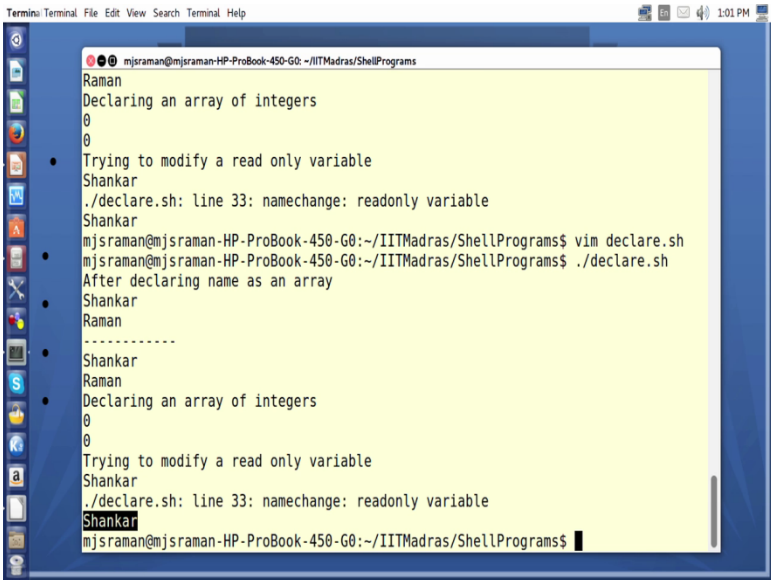

Terminal Terminal File Edit View Search Terminal Help
1:01 PM
mjsraman@mjsraman-HP-ProBook-450-G0: ~/IITMadras/ShellPrograms

```
13 echo $i
14 done
15
16
17 echo "Declaring an array of integers"
18 declare -ai names
19 names[0]="Shankar"
20 names[1]="Raman"
21
22
23 for i in "${names[@]}"
24 do
25 echo $i
26 done
27
28 echo "Trying to modify a read only variable"
29 namechange="Shankar"
30 declare -r namechange
31 echo $namechange
32
33 namechange="Raman"
34 echo $namechange
35
:
```

In the next step ok, so I am trying to modify a variable ok as a read only, so here what I am trying to do is, I am putting a variable name change and then what I am doing is I am declaring that variable to be minus r which means essentially the name change variable only be read only, now what you'll think will happen if I put interchange line number 30 and 29, so for example I put 30 first and then I put line 29, now essentially what will happen is that it not allow you to declare the variable because you already making the variable read only,

So you've to first as in the value of the variable read only, so you've to first assign the value of the variable and then fix its name as read only and then I can print the value now if I want to change the value of the name ok, let's see what happens ok. So what I can do is I can just do it like this, so what I can do is I can print the name change, the value of name change and then I can assign a new value to this variable ok, and then I can see what happens so let's try to do this.

(Refer Slide Time: 11:13)

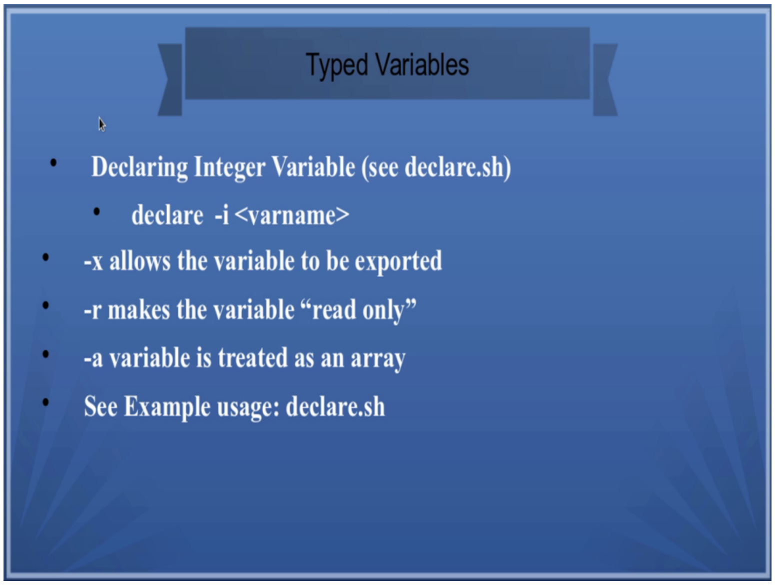


Termina Terminal File Edit View Search Terminal Help
1:01 PM
mjsraman@mjsraman-HP-ProBook-450-G0: ~/IITMadras/ShellPrograms

```
Raman
Declaring an array of integers
0
0
Trying to modify a read only variable
Shankar
./declare.sh: line 33: namechange: readonly variable
Shankar
mjsraman@mjsraman-HP-ProBook-450-G0:~/IITMadras/ShellPrograms$ vim declare.sh
mjsraman@mjsraman-HP-ProBook-450-G0:~/IITMadras/ShellPrograms$ ./declare.sh
After declaring name as an array
Shankar
Raman
------------
Shankar
Raman
Declaring an array of integers
0
0
Trying to modify a read only variable
Shankar
./declare.sh: line 33: namechange: readonly variable
Shankar
mjsraman@mjsraman-HP-ProBook-450-G0:~/IITMadras/ShellPrograms$
```

So I am just declaring this ok, so if you look at this it just said it assign the value of Shankar to the variable and then after that what happen was when I tried to change the name of the name change variable to Raman, it just clearly said that it's only a read only variable. Now this is a sort of a protection that is given to you when you are coding ok, so if you want to ensure that your type I mean this is known as strongly type ok and essentially what happens is if you see the name doesn't get changed.

(Refer Slide Time: 11:50)

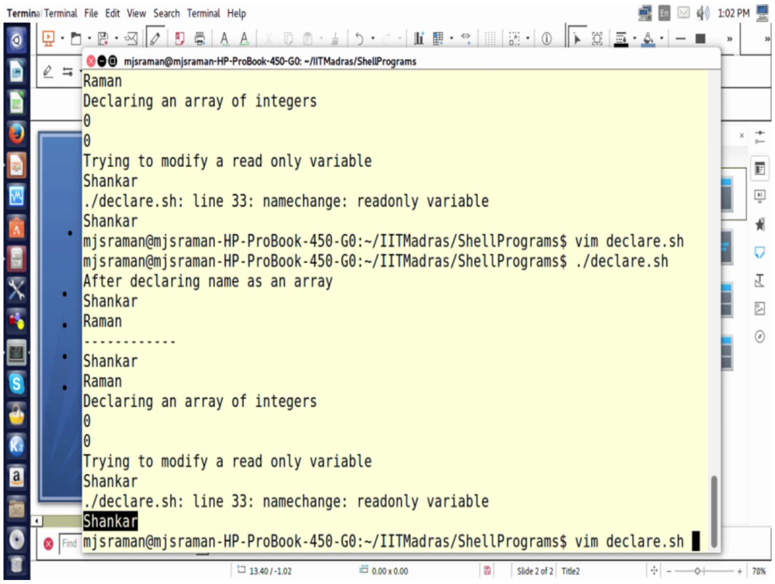


Typed Variables

- Declaring Integer Variable (see declare.sh)
  - declare -i <varname>
- -x allows the variable to be exported
- -r makes the variable “read only”
- -a variable is treated as an array
- See Example usage: declare.sh

So by using this declares statements what you could do is you could ensure that ok your variables are, Sort of type protected, ok?

(Refer Slide Time: 12:03)

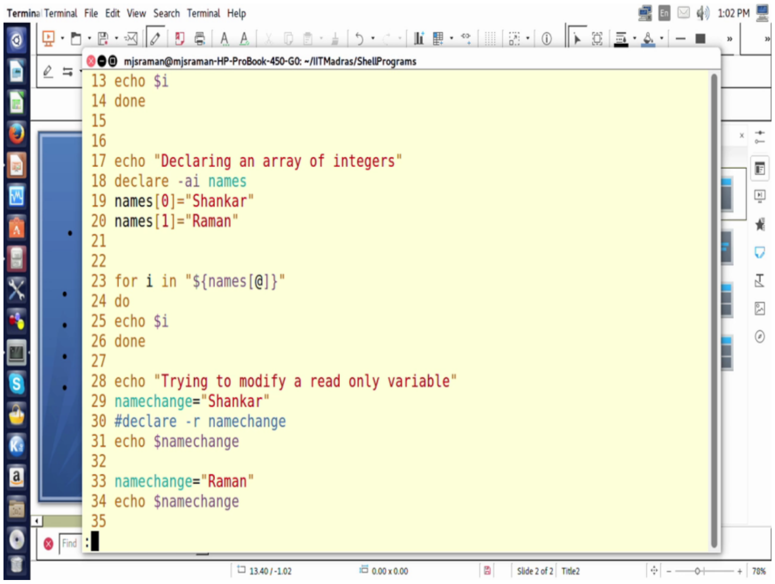


```
Raman
Declaring an array of integers
0
0
Trying to modify a read only variable
Shankar
./declare.sh: line 33: namechange: readonly variable
Shankar
mjsraman@mjsraman-HP-ProBook-450-G0:~/IITMadras/ShellPrograms$ vim declare.sh
mjsraman@mjsraman-HP-ProBook-450-G0:~/IITMadras/ShellPrograms$ ./declare.sh
After declaring name as an array
Shankar
Raman
------------
Shankar
Raman
Declaring an array of integers
0
0
Trying to modify a read only variable
Shankar
./declare.sh: line 33: namechange: readonly variable
Shankar
mjsraman@mjsraman-HP-ProBook-450-G0:~/IITMadras/ShellPrograms$ vim declare.sh
```

mjsraman@mjsraman-HP-ProBook-450-G0: ~/IITMadras/ShellPrograms
Terminal Terminal File Edit View Search Terminal Help
1:02 PM
Slide 2 of 2
78%

So what we can do is, ok? Finally to show you that yes this really has an effect what I’ll try to do is I’ll just go ahead and then change this, Ok

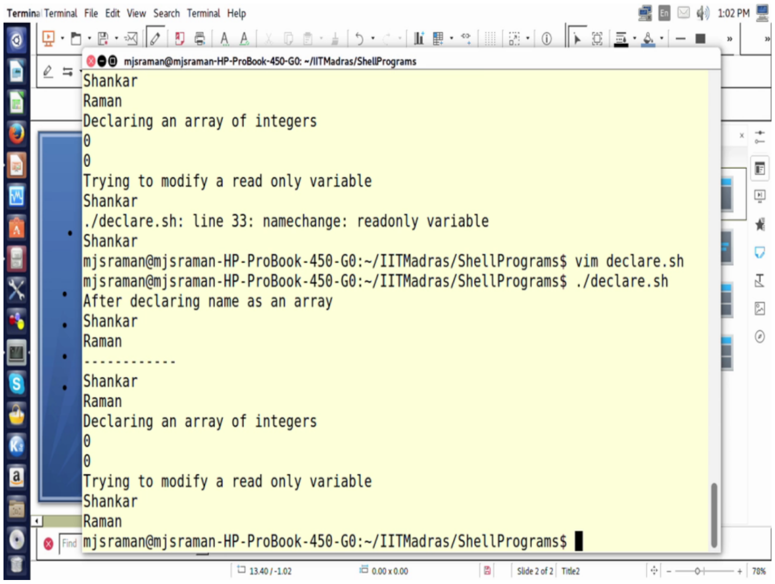
(Refer Slide Time: 12:16)



```
13 echo $i
14 done
15
16
17 echo "Declaring an array of integers"
18 declare -ai names
19 names[0]="Shankar"
20 names[1]="Raman"
21
22
23 for i in "${names[@]}"
24 do
25 echo $i
26 done
27
28 echo "Trying to modify a read only variable"
29 namechange="Shankar"
30 #declare -r namechange
31 echo $namechange
32
33 namechange="Raman"
34 echo $namechange
35
:
```

So let me now in the last line so instead of declaring it as read only I just command this out ok,

(Refer Slide Time: 12:24)



```
Shankar
Raman
Declaring an array of integers
0
0
Trying to modify a read only variable
Shankar
./declare.sh: line 33: namechange: readonly variable
Shankar
mjsraman@mjsraman-HP-ProBook-450-G0:~/IITMadras/ShellPrograms$ vim declare.sh
mjsraman@mjsraman-HP-ProBook-450-G0:~/IITMadras/ShellPrograms$ ./declare.sh
After declaring name as an array
Shankar
Raman
------------
Shankar
Raman
Declaring an array of integers
0
0
Trying to modify a read only variable
Shankar
Raman
mjsraman@mjsraman-HP-ProBook-450-G0:~/IITMadras/ShellPrograms$
```

Now let’s see whether it tries to works, so if you look at this, now the variable name has got change. So in this way you’ll be able to protect your variables in your programs in shell scripting. Ok, this is one of the very important features of shell scripting, so we thought you should understand about this feature, Thank you very much.