

Mr. C gets Interactive

ESC101: Foundations of Computing

Purushottam Kar

Announcements - Lab

- Please go to lab only on the day of your section, be on time
 - *Section-day allotment same as before (please check course website)*
 - *Contact DoAA office in case section not allotted etc*
- Please bring your IITK ID card (no PAN, Aadhar card) to lab
- Set your proxy to relproxy before lab starts – for minor quiz
- Piazza will be deactivated during lab hours Mon-Thu 2-5PM
- Only handwritten notes allowed in lab (no photocopies, slides etc)
- Do not browse internet during lab (YouTube, Google, StackExchange)
 - *Will be considered cheating*
 - *In case of doubt, as TA, tutor, instructor*



Announcements - Piazza

- If not enrolled on Piazza or not activated account, please do so asap
 - *You will miss out on useful discussions, hints, tips*
 - *May also miss out on important announcements*
- How to use Piazza
 - *Please read etiquettes on course website*
 - *Not a good thing to give out solutions (Piazza not a place to show off)*
 - *When asking question*
 - Please check if it has been already answered
 - Websites like StackExchange may ban you if you do not do this ☺
 - Please give details of what went wrong
 - “Why is this not working” is not at all helpful for someone trying to help you



Fun with Integers

4



Fun with Integers

4

Operation	C Code	a	b	c
Addition	<code>c = a + b;</code>	5	4	9
Subtraction	<code>c = a - b;</code>	4	5	-1
Multiplication	<code>c = a * b;</code>	-2	-4	8
Division	<code>c = a / b;</code>	7	2	3
Remainder	<code>c = a % b;</code>	7	2	1



Fun with Integers

4

Operation	C Code	a	b	c
Addition	<code>c = a + b;</code>	5	4	9
Subtraction	<code>c = a - b;</code>	4	5	-1
Multiplication	<code>c = a * b;</code>	-2	-4	8
Division	<code>c = a / b;</code>	7	2	3
Remainder	<code>c = a % b;</code>	7	2	1
Bracketing	<code>c = (a+b)/2;</code>	5	1	3
	<code>c = a + b/2;</code>	5	1	5



Fun with Integers

4

Operation	C Code	a	b	c
Addition	<code>c = a + b;</code>	5	4	9
Subtraction	<code>c = a - b;</code>	4	5	-1
Multiplication	<code>c = a * b;</code>	-2	-4	8
Division	<code>c = a / b;</code>	7	2	3
Remainder	<code>c = a % b;</code>	7	2	1
Bracketing	<code>c = (a+b)/2;</code>	5	1	3
	<code>c = a + b/2;</code>	5	1	5

Recall your BODMAS rules from high school



Fun with Integers

4

Operation	C Code	a	b	c
Addition	<code>c = a + b;</code>	5	4	9
Subtraction	<code>c = a - b;</code>	4	5	-1
Multiplication	<code>c = a * b;</code>	-2	-4	8
Division	<code>c = a / b;</code>	7	2	3
Remainder	<code>c = a % b;</code>	7	2	1
Bracketing	<code>c = (a+b)/2;</code>	5	1	3
	<code>c = a + b/2;</code>	5	1	5

Recall your BODMAS rules from high school

Mr. C follows similar rules – will see in detail soon



Fun with Integers

4

Operation	C Code	a	b	c
Addition	<code>c = a + b;</code>	5	4	9
Subtraction	<code>c = a - b;</code>	4	5	-1
Multiplication	<code>c = a * b;</code>	-2	-4	8
Division	<code>c = a / b;</code>	7	2	3
Remainder	<code>c = a % b;</code>	7	2	1
Bracketing	<code>c = (a+b)/2;</code>	5	1	3
	<code>c = a + b/2;</code>	5	1	5

Recall your BODMAS rules from high school

Mr. C follows similar rules – will see in detail soon

Good practice to bracket your formulae



Fun with Integers

4

Operation	C Code	a	b	c
Addition	<code>c = a + b;</code>	5	4	9
Subtraction	<code>c = a - b;</code>	4	5	-1
Multiplication	<code>c = a * b;</code>	-2	-4	8
Division	<code>c = a / b;</code>	7	2	3
Remainder	<code>c = a % b;</code>	7	2	1
Bracketing	<code>c = (a+b)/2;</code>	5	1	3
	<code>c = a + b/2;</code>	5	1	5

Recall your BODMAS rules from high school

Mr. C follows similar rules – will see in detail soon

Good practice to bracket your formulae

Minimize confusion as well as chances of error



Fun with Integers

4

Operation	C Code	a	b	c
Addition	<code>c = a + b;</code>	5	4	9
Subtraction	<code>c = a - b;</code>	4	5	-1
Multiplication	<code>c = a * b;</code>	-2	-4	8
Division	<code>c = a / b;</code>	7	2	3
Remainder	<code>c = a % b;</code>	7	2	1
Bracketing	<code>c = (a+b)/2;</code>	5	1	3
	<code>c = a + b/2;</code>	5	1	5

Recall your BODMAS rules from high school

Mr. C follows similar rules – will see in detail soon

Good practice to bracket your formulae

Minimize confusion as well as chances of error

Play with brackets in lab to practice



A handy shortcut

12



A handy shortcut

12

HOW WE USUALLY SPEAK TO A HUMAN



A handy shortcut

12

HOW WE USUALLY SPEAK TO A HUMAN



A handy shortcut

12

HOW WE USUALLY SPEAK TO A HUMAN

Do you speak English?



A handy shortcut

12

HOW WE USUALLY SPEAK TO A HUMAN

Do you speak English?
Hello



A handy shortcut

12

HOW WE USUALLY SPEAK TO A HUMAN

Do you speak English?

Hello

a,b,c are variables.



A handy shortcut

12

HOW WE USUALLY SPEAK TO A HUMAN

Do you speak English?

Hello

a,b,c are variables.

$a = 5$ and $b = 4$.



A handy shortcut

12

HOW WE USUALLY SPEAK TO A HUMAN

Do you speak English?

Hello

a,b,c are variables.

$a = 5$ and $b = 4$.

Please add them and put the result in variable c.



A handy shortcut

12

HOW WE USUALLY SPEAK TO A HUMAN

Do you speak English?

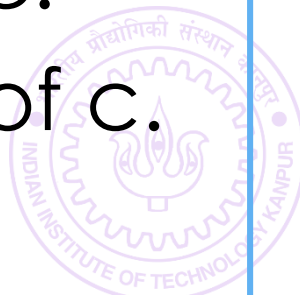
Hello

a,b,c are variables.

$a = 5$ and $b = 4$.

Please add them and put the result in variable c.

Please tell me value of c.



A handy shortcut

12

HOW WE USUALLY SPEAK TO A HUMAN

Do you speak English?

Hello

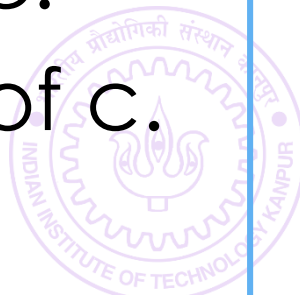
a,b,c are variables.

$a = 5$ and $b = 4$.

Please add them and put the result in variable c.

Please tell me value of c.

Goodbye



A handy shortcut

12

HOW WE MUST SPEAK TO MR. COMPILER

HOW WE USUALLY SPEAK TO A HUMAN

Do you speak English?

Hello

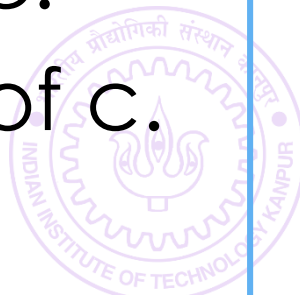
a,b,c are variables.

a = 5 and b = 4.

Please add them and put the result in variable c.

Please tell me value of c.

Goodbye



A handy shortcut

12

HOW WE MUST SPEAK TO MR. COMPILER

HOW WE USUALLY SPEAK TO A HUMAN

Do you speak English?

Hello

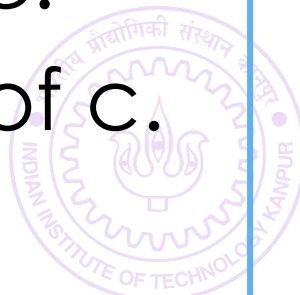
a,b,c are variables.

a = 5 and b = 4.

Please add them and put the result in variable c.

Please tell me value of c.

Goodbye



A handy shortcut

12

HOW WE MUST SPEAK TO MR. COMPILER

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

HOW WE USUALLY SPEAK TO A HUMAN

Do you speak English?

Hello

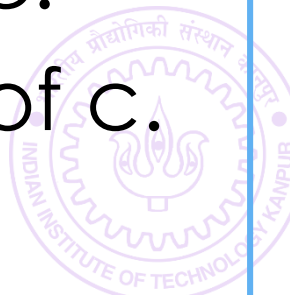
a,b,c are variables.

a = 5 and b = 4.

Please add them and put the result in variable c.

Please tell me value of c.

Goodbye



A handy shortcut

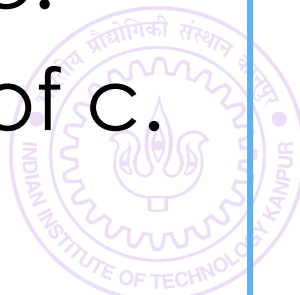
12

HOW WE MUST SPEAK TO MR. COMPILER

```
#include<stdio.h>  
int main(){
```

HOW WE USUALLY SPEAK TO A HUMAN

Do you speak English?
Hello
a,b,c are variables.
a = 5 and b = 4.
Please add them and put
the result in variable c.
Please tell me value of c.
Goodbye



A handy shortcut

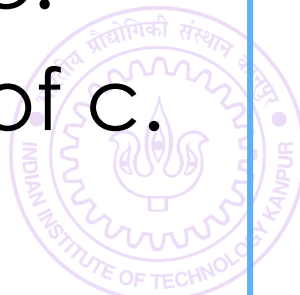
12

HOW WE MUST SPEAK TO MR. COMPILER

```
#include<stdio.h>
int main(){
int a, b, c;
```

HOW WE USUALLY SPEAK TO A HUMAN

Do you speak English?
Hello
a,b,c are variables.
a = 5 and b = 4.
Please add them and put
the result in variable c.
Please tell me value of c.
Goodbye



A handy shortcut

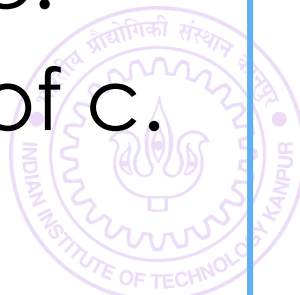
12

HOW WE MUST SPEAK TO MR. COMPILER

```
#include<stdio.h>
int main(){
int a, b, c;
a = 5, b = 4;
```

HOW WE USUALLY SPEAK TO A HUMAN

Do you speak English?
Hello
a,b,c are variables.
a = 5 and b = 4.
Please add them and put
the result in variable c.
Please tell me value of c.
Goodbye



A handy shortcut

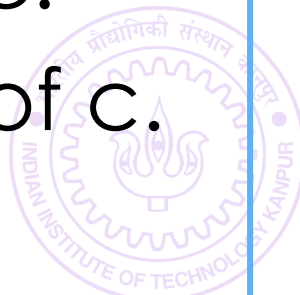
12

HOW WE MUST SPEAK TO MR. COMPILER

```
#include<stdio.h>
int main(){
int a, b, c;
a = 5, b = 4;
c = a + b;
```

HOW WE USUALLY SPEAK TO A HUMAN

Do you speak English?
Hello
a,b,c are variables.
a = 5 and b = 4.
Please add them and put
the result in variable c.
Please tell me value of c.
Goodbye



A handy shortcut

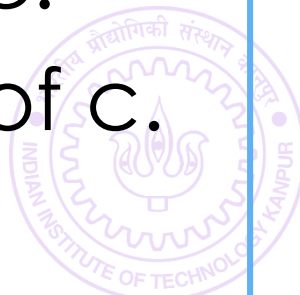
12

HOW WE MUST SPEAK TO MR. COMPILER

```
#include<stdio.h>
int main(){
int a, b, c;
a = 5, b = 4;
c = a + b;
printf("%d",c);
```

HOW WE USUALLY SPEAK TO A HUMAN

Do you speak English?
Hello
a,b,c are variables.
a = 5 and b = 4.
Please add them and put
the result in variable c.
Please tell me value of c.
Goodbye



A handy shortcut

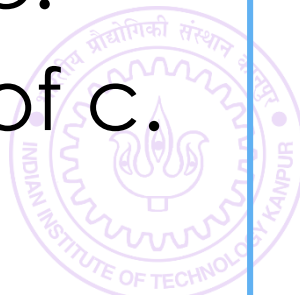
12

HOW WE MUST SPEAK TO MR. COMPILER

```
#include<stdio.h>
int main(){
int a, b, c;
a = 5, b = 4;
c = a + b;
printf("%d",c);
return 0;
```

HOW WE USUALLY SPEAK TO A HUMAN

Do you speak English?
Hello
a,b,c are variables.
a = 5 and b = 4.
Please add them and put
the result in variable c.
Please tell me value of c.
Goodbye



A handy shortcut

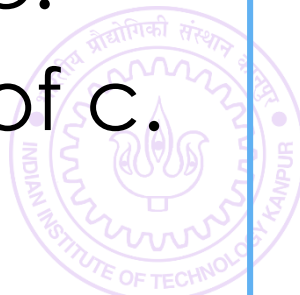
12

HOW WE MUST SPEAK TO MR. COMPILER

```
#include<stdio.h>
int main(){
int a, b, c;
a = 5, b = 4;
c = a + b;
printf("%d",c);
return 0;
}
```

HOW WE USUALLY SPEAK TO A HUMAN

Do you speak English?
Hello
a,b,c are variables.
a = 5 and b = 4.
Please add them and put
the result in variable c.
Please tell me value of c.
Goodbye

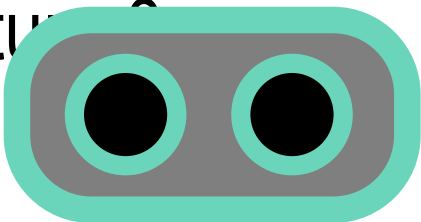


A handy shortcut

12

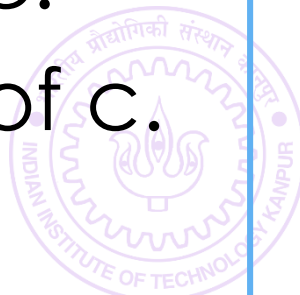
HOW WE MUST SPEAK TO MR. COMPILER

```
#include<stdio.h>
int main(){
int a, b, c;
a = 5, b = 4;
c = a + b;
printf("%d",c);
return 0;
}
```



HOW WE USUALLY SPEAK TO A HUMAN

Do you speak English?
Hello
a,b,c are variables.
a = 5 and b = 4.
Please add them and put
the result in variable c.
Please tell me value of c.
Goodbye



A handy shortcut

12

HOW WE MUST SPEAK TO MR. COMPILER

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

```
int main(){
```

```
int a, b, c;
```

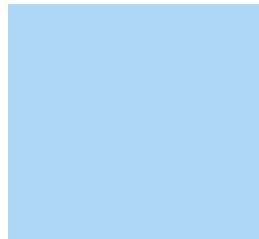
```
a = 5, b = 4;
```

```
c = a + b;
```

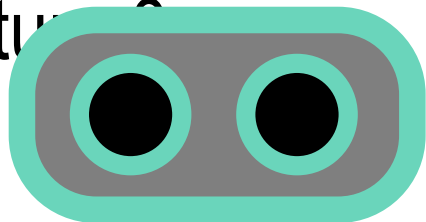
```
printf("%d",c);
```

```
return 0;
```

```
}
```



a



HOW WE USUALLY SPEAK TO A HUMAN

Do you speak English?

Hello

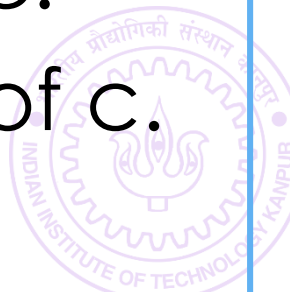
a,b,c are variables.

a = 5 and b = 4.

Please add them and put the result in variable c.

Please tell me value of c.

Goodbye



A handy shortcut

12

HOW WE MUST SPEAK TO MR. COMPILER

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

```
int main(){
```

```
int a, b, c;
```

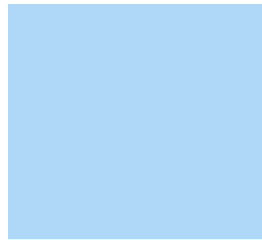
```
a = 5, b = 4;
```

```
c = a + b;
```

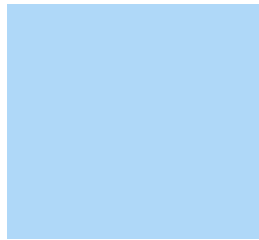
```
printf("%d",c);
```

```
return 0;
```

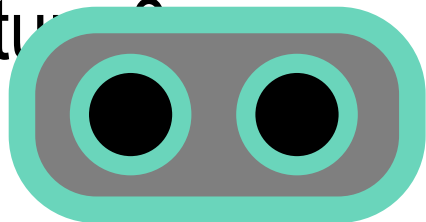
```
}
```



a



b



HOW WE USUALLY SPEAK TO A HUMAN

Do you speak English?

Hello

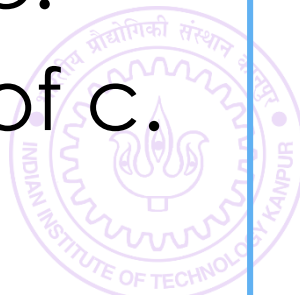
a,b,c are variables.

a = 5 and b = 4.

Please add them and put the result in variable c.

Please tell me value of c.

Goodbye



A handy shortcut

12

HOW WE MUST SPEAK TO MR. COMPILER

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

```
int main(){
```

```
int a, b, c;
```

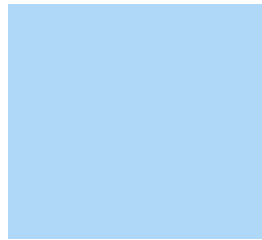
```
a = 5, b = 4;
```

```
c = a + b;
```

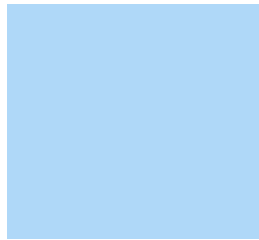
```
printf("%d",c);
```

```
return 0;
```

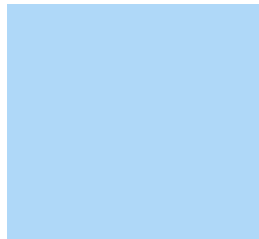
```
}
```



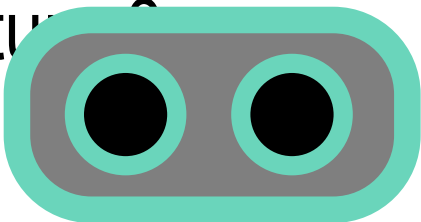
a



b



c



HOW WE USUALLY SPEAK TO A HUMAN

Do you speak English?

Hello

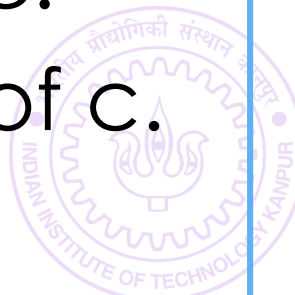
a,b,c are variables.

a = 5 and b = 4.

Please add them and put the result in variable c.

Please tell me value of c.

Goodbye



A handy shortcut

12

HOW WE MUST SPEAK TO MR. COMPILER

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

```
int main(){
```

```
int a, b, c;
```

```
a = 5, b = 4;
```

```
c = a + b;
```

```
printf("%d",c);
```

```
return 0;
```

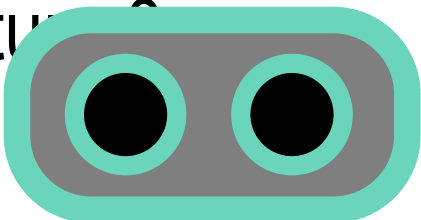
```
}
```

5

a

b

c



HOW WE USUALLY SPEAK TO A HUMAN

Do you speak English?

Hello

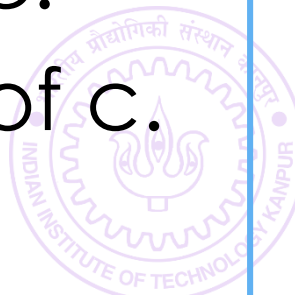
a,b,c are variables.

a = 5 and b = 4.

Please add them and put the result in variable c.

Please tell me value of c.

Goodbye



A handy shortcut

12

HOW WE MUST SPEAK TO MR. COMPILER

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

```
int main(){
```

```
int a, b, c;
```

```
a = 5, b = 4;
```

```
c = a + b;
```

```
printf("%d",c);
```

```
return 0;
```

```
}
```

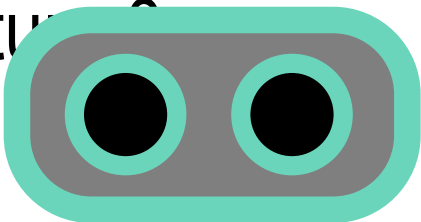
5

a

4

b

c



HOW WE USUALLY SPEAK TO A HUMAN

Do you speak English?

Hello

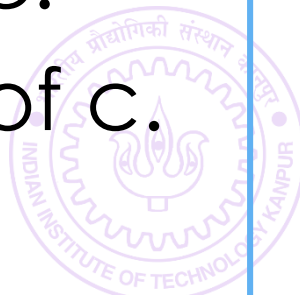
a,b,c are variables.

a = 5 and b = 4.

Please add them and put the result in variable c.

Please tell me value of c.

Goodbye



A handy shortcut

12

HOW WE MUST SPEAK TO MR. COMPILER

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

```
int main(){
```

```
int a, b, c;
```

```
a = 5, b = 4;
```

```
c = a + b;
```

```
printf("%d",c);
```

```
return 0;
```

```
}
```

5

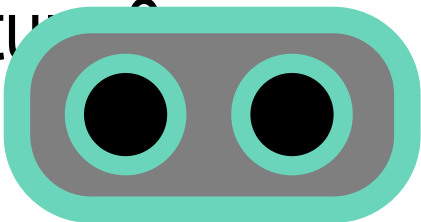
a

4

b

9

c



HOW WE USUALLY SPEAK TO A HUMAN

Do you speak English?

Hello

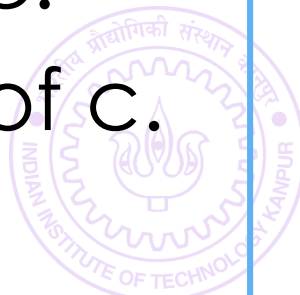
a,b,c are variables.

a = 5 and b = 4.

Please add them and put the result in variable c.

Please tell me value of c.

Goodbye



A handy shortcut

12

HOW WE MUST SPEAK TO MR. COMPILER

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

```
int main(){
```

```
int a, b, c;
```

```
a = 5, b = 4;
```

```
c = a + b;
```

```
printf("%d",c);
```

```
return 0;
```

```
}
```

5

a

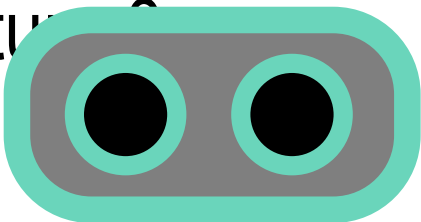
9

4

b

9

c



HOW WE USUALLY SPEAK TO A HUMAN

Do you speak English?

Hello

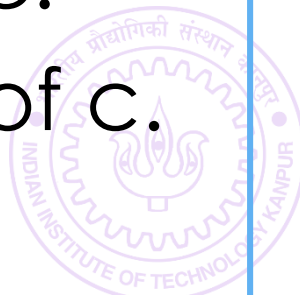
a,b,c are variables.

a = 5 and b = 4.

Please add them and put the result in variable c.

Please tell me value of c.

Goodbye



A handy shortcut

12

HOW WE MUST SPEAK TO MR. COMPILER

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

```
int main(){
```

```
int a, b, c;
```

```
a = 5, b = 4;
```

```
c = a + b;
```

```
printf("%d",c);
```

```
return 0;
```

```
}
```

5

a

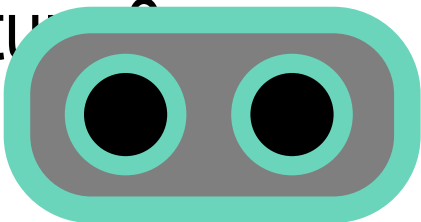
4

b

9

9

c



HOW WE USUALLY SPEAK TO A HUMAN

A handy shortcut

12

HOW WE MUST SPEAK TO MR. COMPILER

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

```
int main(){
```

```
int a, b, c;
```

```
a = 5, b = 4;
```

```
c = a + b;
```

```
printf("%d",c);
```

```
return 0;
```

```
}
```

5

a

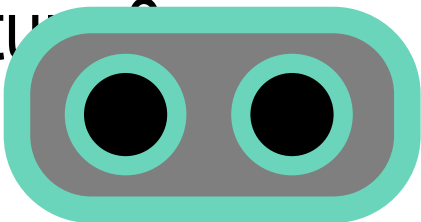
4

b

9

9

c



HOW WE USUALLY SPEAK TO A HUMAN

Do you speak English?

A handy shortcut

12

HOW WE MUST SPEAK TO MR. COMPILER

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

```
int main(){
```

```
int a, b, c;
```

```
a = 5, b = 4;
```

```
c = a + b;
```

```
printf("%d",c);
```

```
return 0;
```

```
}
```

5

a

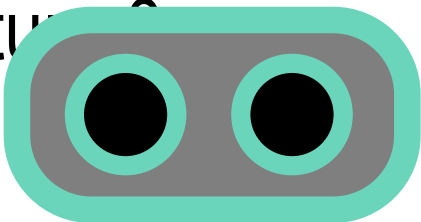
9

4

b

9

c



HOW WE USUALLY SPEAK TO A HUMAN

Do you speak English?

Hello

A handy shortcut

12

HOW WE MUST SPEAK TO MR. COMPILER

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

```
int main(){
```

```
int a, b, c;
```

```
a = 5, b = 4;
```

```
c = a + b;
```

```
printf("%d",c);
```

```
return 0;
```

```
}
```

5

a

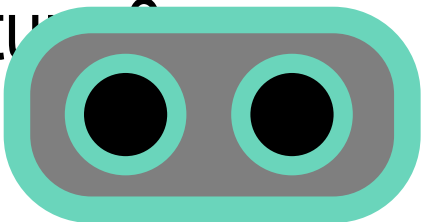
4

b

9

9

c



HOW WE USUALLY SPEAK TO A HUMAN

Do you speak English?

Hello

a,b are integer variables.

A handy shortcut

12

HOW WE MUST SPEAK TO MR. COMPILER

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

```
int main(){
```

```
int a, b, c;
```

```
a = 5, b = 4;
```

```
c = a + b;
```

```
printf("%d",c);
```

```
return 0;
```

```
}
```

5

a

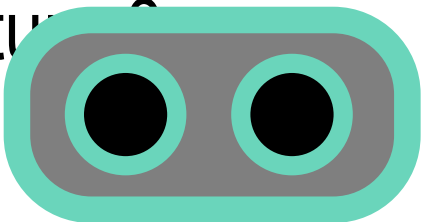
4

b

9

9

c



HOW WE USUALLY SPEAK TO A HUMAN

Do you speak English?

Hello

a,b are integer variables.

a = 5 and b = 4.

A handy shortcut

12

HOW WE MUST SPEAK TO MR. COMPILER

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

```
int main(){
```

```
int a, b, c;
```

```
a = 5, b = 4;
```

```
c = a + b;
```

```
printf("%d",c);
```

```
return 0;
```

```
}
```

5

a

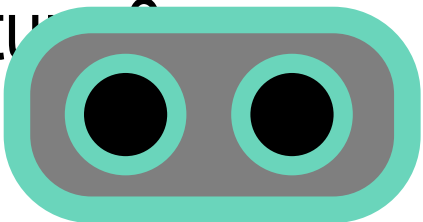
4

b

9

9

c



HOW WE USUALLY SPEAK TO A HUMAN

Do you speak English?

Hello

a,b are integer variables.

a = 5 and b = 4.

Please add tell me their sum.

A handy shortcut

12

HOW WE MUST SPEAK TO MR. COMPILER

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

```
int main(){
```

```
int a, b, c;
```

```
a = 5, b = 4;
```

```
c = a + b;
```

```
printf("%d",c);
```

```
return 0;
```

```
}
```

5

a

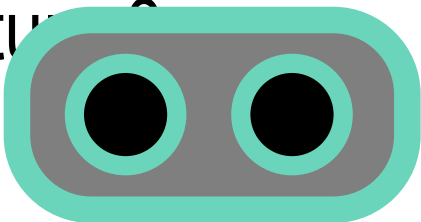
4

b

9

9

c



HOW WE USUALLY SPEAK TO A HUMAN

Do you speak English?

Hello

a,b are integer variables.

a = 5 and b = 4.

Please add tell me their sum.

Goodbye

A handy shortcut

12

HOW WE MUST SPEAK TO MR. COMPILER

```
#include<stdio.h>
int main(){
int a, b;
a = 5, b = 4;
printf("%d",a+b);
return 0;
}
```

HOW WE USUALLY SPEAK TO A HUMAN

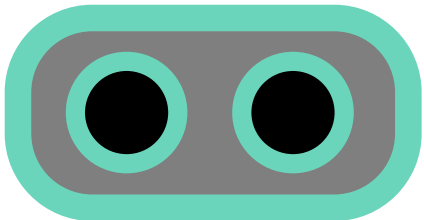
Do you speak English?
Hello
a,b are integer variables.
a = 5 and b = 4.
Please add tell me their
sum.
Goodbye

A handy shortcut

12

HOW WE MUST SPEAK TO MR. COMPILER

```
#include<stdio.h>
int main(){
int a, b;
a = 5, b = 4;
printf("%d",a+b);
return 0;
}
```



HOW WE USUALLY SPEAK TO A HUMAN

Do you speak English?
Hello
a,b are integer variables.
a = 5 and b = 4.
Please add tell me their
sum.
Goodbye

A handy shortcut

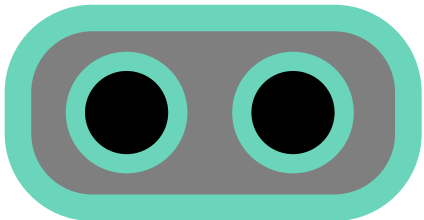
12

HOW WE MUST SPEAK TO MR. COMPILER

```
#include<stdio.h>
int main(){
int a, b;
a = 5, b = 4;
printf("%d",a+b);
return 0;
}
```



a



HOW WE USUALLY SPEAK TO A HUMAN

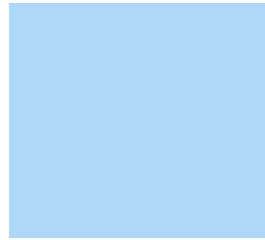
Do you speak English?
Hello
a,b are integer variables.
a = 5 and b = 4.
Please add tell me their
sum.
Goodbye

A handy shortcut

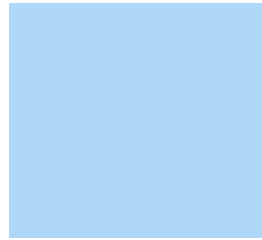
12

HOW WE MUST SPEAK TO MR. COMPILER

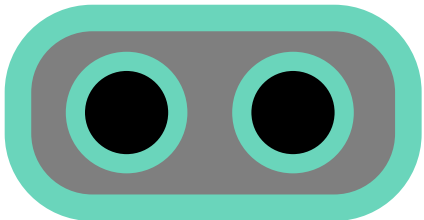
```
#include<stdio.h>
int main(){
int a, b;
a = 5, b = 4;
printf("%d",a+b);
return 0;
}
```



a



b



HOW WE USUALLY SPEAK TO A HUMAN

Do you speak English?
Hello
a,b are integer variables.
a = 5 and b = 4.
Please add tell me their
sum.
Goodbye

A handy shortcut

12

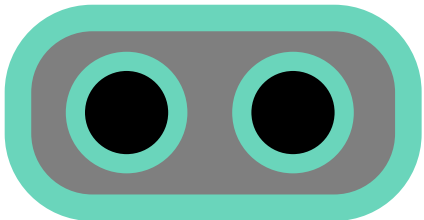
HOW WE MUST SPEAK TO MR. COMPILER

```
#include<stdio.h>
int main(){
int a, b;
a = 5, b = 4;
printf("%d",a+b);
return 0;
}
```

5

a

b



HOW WE USUALLY SPEAK TO A HUMAN

Do you speak English?
Hello
a,b are integer variables.
a = 5 and b = 4.
Please add tell me their
sum.
Goodbye

A handy shortcut

12

HOW WE MUST SPEAK TO MR. COMPILER

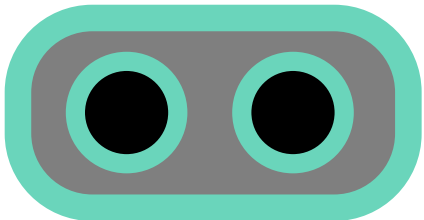
```
#include<stdio.h>
int main(){
int a, b;
a = 5, b = 4;
printf("%d",a+b);
return 0;
}
```

5

a

4

b



HOW WE USUALLY SPEAK TO A HUMAN

Do you speak English?
Hello
a,b are integer variables.
a = 5 and b = 4.
Please add tell me their
sum.
Goodbye

A handy shortcut

12

HOW WE MUST SPEAK TO MR. COMPILER

```
#include<stdio.h>
int main(){
int a, b;
a = 5, b = 4;
printf("%d",a+b);
return 0;
}
```

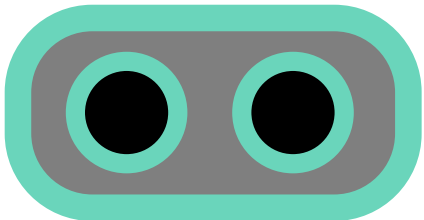
5

a

4

b

9



HOW WE USUALLY SPEAK TO A HUMAN

Do you speak English?
Hello
a,b are integer variables.
a = 5 and b = 4.
Please add tell me their
sum.
Goodbye

A handy shortcut

12

HOW WE MUST SPEAK TO MR. COMPILER

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

```
int main(){
```

```
int a, b;
```

```
a = 5, b = 4;
```

```
printf("%d",a+b);
```

```
return 0;
```

```
}
```

5

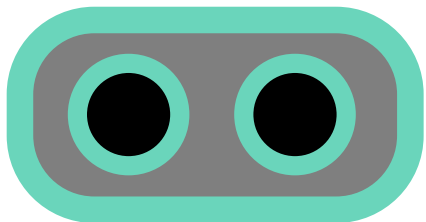
a

4

b

9

9



HOW WE USUALLY SPEAK TO A HUMAN

Do you speak English?

Hello

a,b are integer variables.

a = 5 and b = 4.

Please add tell me their sum.

Goodbye

A handy shortcut

12

HOW WE MUST SPEAK TO MR. COMPILER

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

```
int main(){
```

```
int a, b;
```

```
a = 5, b = 4;
```

```
printf("%d",a+b);
```

```
return 0;
```

```
}
```

5

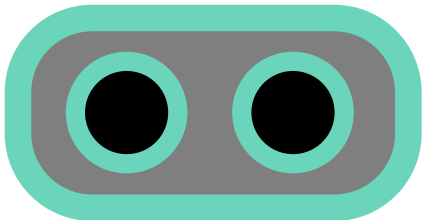
a

4

b

9

9



HOW WE USUALLY SPEAK TO A HUMAN

Do you speak English?

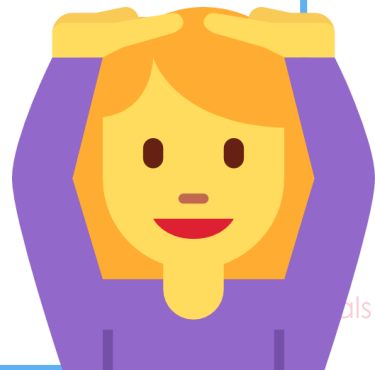
Hello

a,b are integer variables.

a = 5 and b = 4.

Please add tell me their sum.

Goodbye



A handy shortcut

12

HOW WE MUST SPEAK TO MR. COMPILER

HOW WE USUALLY SPEAK TO A HUMAN

```
#include<stdio.h>
int main(){
int a, b;
a = 5, b = 4;
printf("%d",a+b);
return 0;
}
```

5

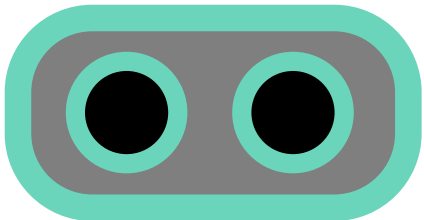
a

4

b

9

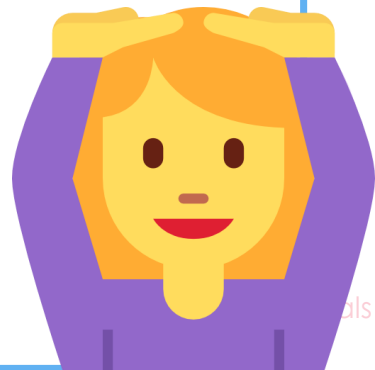
9



Nice! I do not have
to declare variables
unless I really need
them 😊

Please add them and
tell me their
sum.

Goodbye



A handy shortcut

12

HOW WE MUST SPEAK TO MR. COMPILER

HOW WE USUALLY SPEAK TO A HUMAN

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

```
int main(){
```

```
int a, b;
```

```
a = 5, b = 4;
```

```
printf("%d",a+b);
```

```
return 0;
```

```
}
```

5

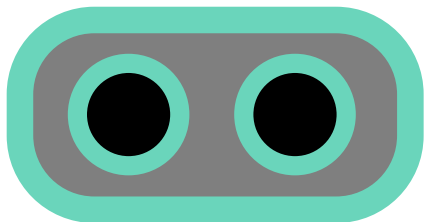
a

4

b

9

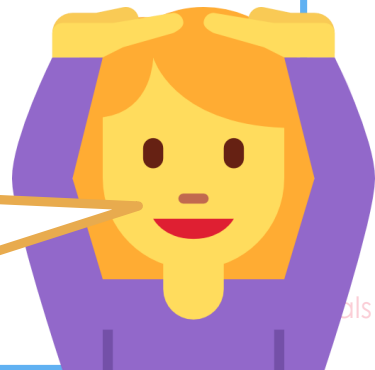
9



Nice! I do not have to declare variables unless I really need them 😊

Please add them for me

Mr C is also not complaining that the value a+b is unused – since it got printed!



A handy tip while solving problems

58



A handy tip while solving problems

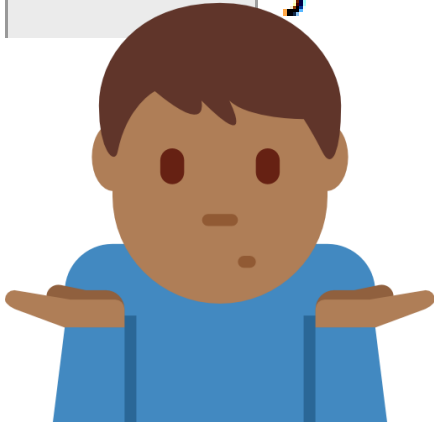
58

```
1 #include<stdio.h>-  
2 int main(){-  
3     int x = 3;-  
4     int result;-  
5     result = 2/3*x*x*x + 2*x*x + 9*x;-  
6     printf("The area under the curve is %d",result);-  
7     return 0;-  
8 }
```



A handy tip while solving problems 58

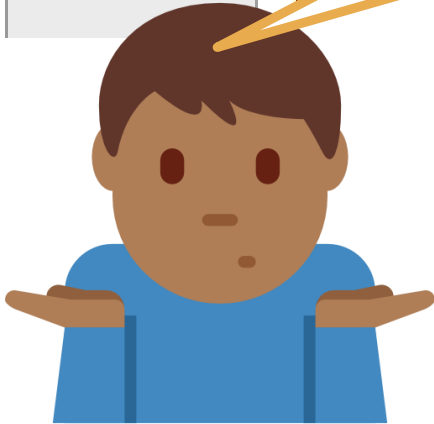
```
1 #include<stdio.h>-\n2 int main(){-\n3     int x = 3;-\n4     int result;-\n5     result = 2/3*x*x*x + 2*x*x + 9*x;-\n6     printf("The area under the curve is %d",result);-\n7     return 0;-\n8 }
```



A handy tip while solving problems 58

```
1 #include<stdio.h>
2 int main(){
3     ... int x = 3;
4     ... int result;
5     ... /3*x*x*x + 2*x*x + 9*x;
6     ... e area under the curve is %d",result);
7
8 }
```

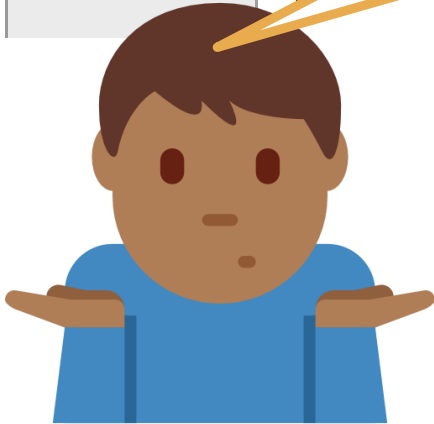
I have no idea
what is going
wrong here!



A handy tip while solving problems 58

```
1 #include<stdio.h>
2 int main(){
3     ... int x = 3;
4     ... int result;
5     ... /3*x*x*x + 2*x*x + 9*x;
6     ... e area under the curve is %d",result);
7
8 }
```

I have no idea
what is going
wrong here!

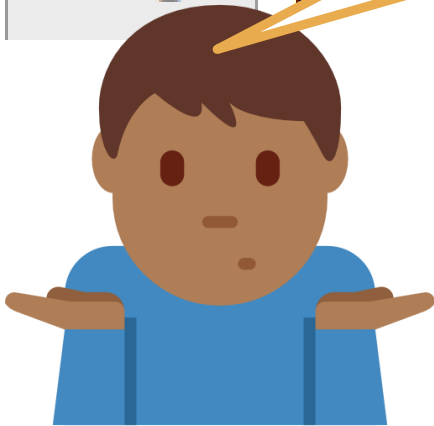


A handy tip while solving problems 58

```
1 #include<stdio.h>
2 int main(){
3     ... int x = 3;
4     ... int result;
5     ... /3*x*x*x + 2*x*x + 9*x;
6     ... e area under the curve is %d", result);
7
8 }
```

I have no idea
what is going
wrong here!

Try breaking up
the problem into
smaller pieces



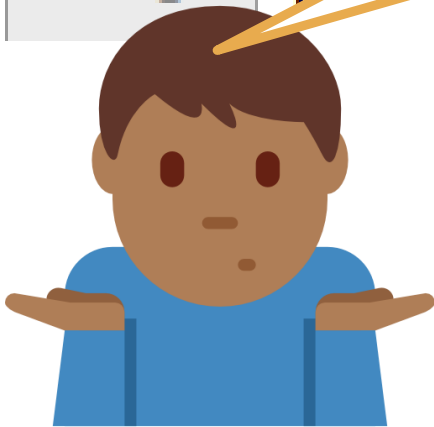
A handy tip while solving problems 58

```
1 #include<stdio.h>
2 int main(){
3     ... int x = 3;
4     ... int result;
5     ... /3*x*x*x + 2*x*x;
6     ... e area under the curve is %d", result);
7
8 }
```

I have no idea
what is going
wrong here!

Print your solutions to
each one of these pieces
to see where going wrong

Try breaking up
the problem into
smaller pieces



Writing pretty code is an art

65



Writing pretty code is an art

65

Last week we learnt about “indentation”



Writing pretty code is an art

65

Last week we learnt about “indentation”

```
int main(){  
    int a;  
    int b;  
    a = 5, b = 4;  
    a + b;  
    return 0;  
}
```

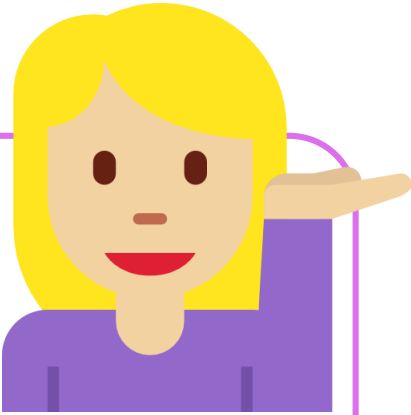


Writing pretty code is an art

65

Last week we learnt about “indentation”

```
int main(){  
    int a;  
    int b;  
    a = 5, b = 4;  
    a + b;  
    return 0;  
}
```



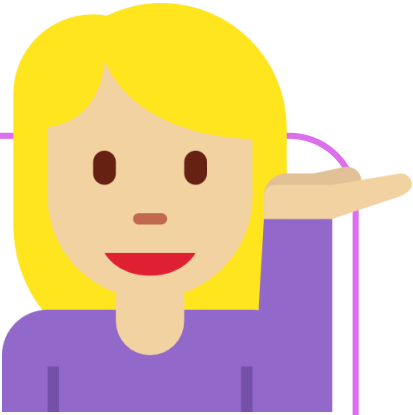
Writing pretty code is an art

65

Last week we learnt about “indentation”

Let us learn about “comments” today

```
int main(){  
    int a;  
    int b;  
    a = 5, b = 4;  
    a + b;  
    return 0;  
}
```



Writing pretty code is an art

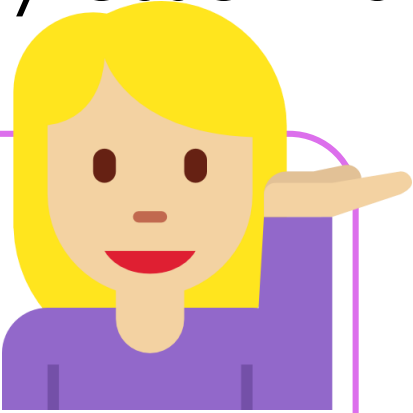
65

Last week we learnt about “indentation”

Let us learn about “comments” today

Absolutely essential in industry, even self projects

```
int main(){  
    int a;  
    int b;  
    a = 5, b = 4;  
    a + b;  
    return 0;  
}
```



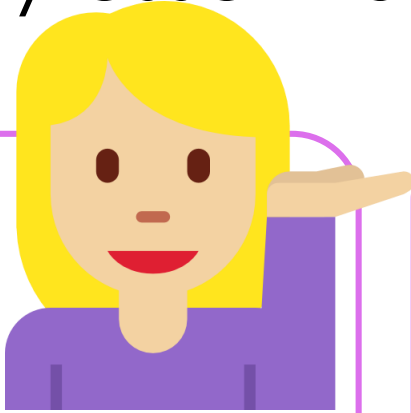
Writing pretty code is an art

65

Last week we learnt about “indentation”

Let us learn about “comments” today

Absolutely essential in industry, even self projects



```
int main(){  
    int a;  
    int b;  
    a = 5, b = 4;  
    a + b;  
    return 0;  
}
```

```
int main(){  
    int a; // My first int  
    int b; // The other int  
    // Assign them values  
    a = 5, b = 4;  
    a + b;  
    return 0;  
}
```



Writing pretty code is an art

65

Last week we learnt about “indentation”

Let us learn about “comments” today

Absolutely essential in industry, even self projects

What we see



```
int main(){  
    int a;  
    int b;  
    a = 5, b = 4;  
    a + b;  
    return 0;  
}
```

```
int main(){  
    int a; // My first int  
    int b; // The other int  
    // Assign them values  
    a = 5, b = 4;  
    a + b;  
    return 0;  
}
```



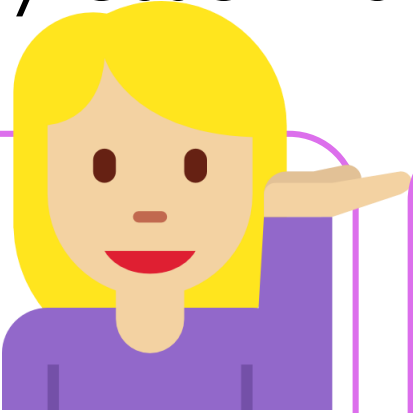
Writing pretty code is an art

65

Last week we learnt about “indentation”

Let us learn about “comments” today

Absolutely essential in industry, even self projects



What we see

```
int main(){  
    int a;  
    int b;  
    a = 5, b = 4;  
    a + b;  
    return 0;  
}
```

```
int main(){  
    int a; // My first int  
    int b; // The other int  
    // Assign them values  
    a = 5, b = 4;  
    a + b;  
    return 0;  
}
```

What Mr C sees



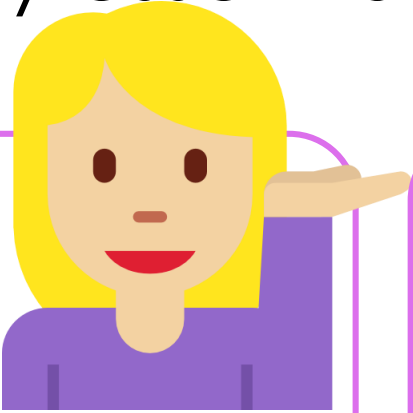
Writing pretty code is an art

65

Last week we learnt about “indentation”

Let us learn about “comments” today

Absolutely essential in industry, even self projects



What we see

```
int main(){  
    int a;  
    int b;  
    a = 5, b = 4;  
    a + b;  
    return 0;  
}
```

```
int main(){  
    int a; // My first int  
    int b; // The other int  
    // Assign them values  
    a = 5, b = 4;  
    a + b;  
    return 0;  
}
```

What Mr C sees

```
int main(){  
    int a;  
    int b;  
  
    a = 5, b = 4;  
    a + b;  
    return 0;  
}
```



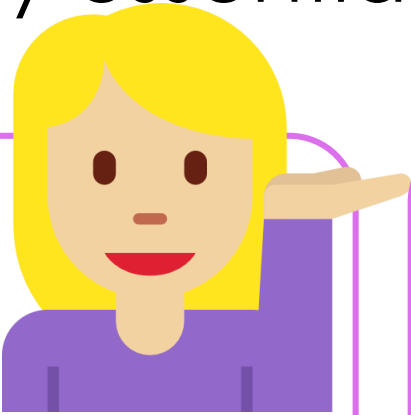
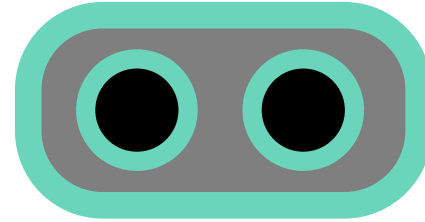
Writing pretty code is an art

65

Last week we learnt about “indentation”

Let us learn about “comments” today

Absolutely essential in industry, even self projects



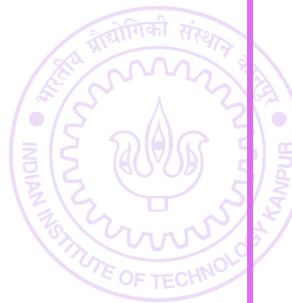
What we see

```
int main(){  
    int a;  
    int b;  
    a = 5, b = 4;  
    a + b;  
    return 0;  
}
```

```
int main(){  
    int a; // My first int  
    int b; // The other int  
    // Assign them values  
    a = 5, b = 4;  
    a + b;  
    return 0;  
}
```

What Mr C sees

```
int main(){  
    int a;  
    int b;  
  
    a = 5, b = 4;  
    a + b;  
    return 0;  
}
```



Writing pretty code is important

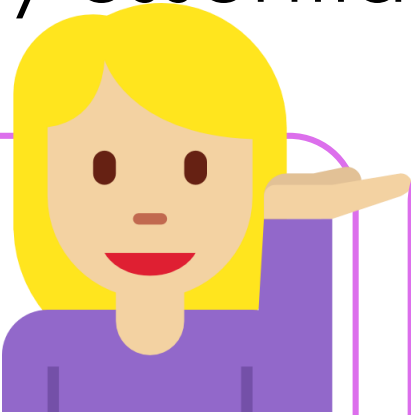
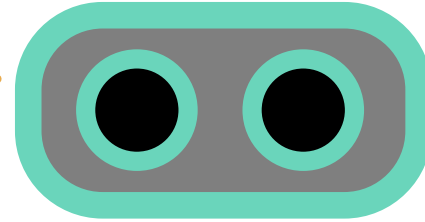
65

Last week we learnt about “indentation”

Let us learn about “comments” today

Absolutely essential in industry, even self projects

Okay. I will
add your two
numbers



What we see

```
int main(){  
    int a;  
    int b;  
    a = 5, b = 4;  
    a + b;  
    return 0;  
}
```

```
int main(){  
    int a; // My first int  
    int b; // The other int  
    // Assign them values  
    a = 5, b = 4;  
    a + b;  
    return 0;  
}
```

What Mr C sees

```
int main(){  
    int a;  
    int b;  
  
    a = 5, b = 4;  
    a + b;  
    return 0;  
}
```



Commenting Styles

77

Since it is an art form, artists differ on what is more pretty



Commenting Styles

77

Since it is an art form, artists differ on what is more pretty

```
int main(){  
    int a; // My first int  
    int b; // The other int  
    // Assign them values  
    a = 5, b = 4;  
    a + b;  
    return 0;  
}
```



Commenting Styles

77

Since it is an art form, artists differ on what is more pretty

```
int main(){  
    int a; // My first int  
    int b; // The other int  
    // Assign them values  
    a = 5, b = 4;  
    a + b;  
    return 0;  
}
```

```
int main(){  
    int a; /* My first int */  
    int b; /* The other int */  
    /* Assign them values */  
    a = 5, b = 4;  
    a + b;  
    return 0;  
}
```



Commenting Styles

77

Since it is an art form, artists differ on what is more pretty

```
int main(){
    int a; // My first int
    int b; // The other int
    // Assign them values
    a = 5, b = 4;
    a + b;
    return 0;
}
```

```
int main(){
    int a; /* My first int */
    int b; /* The other int */
    /* Assign them values */
    a = 5, b = 4;
    a + b;
    return 0;
}
```

```
int main(){
    int a; // My first int
    int b; // The other int
    /* Assign them values */
    a = 5, b = 4;
    a + b;
    return 0;
}
```



Commenting Styles

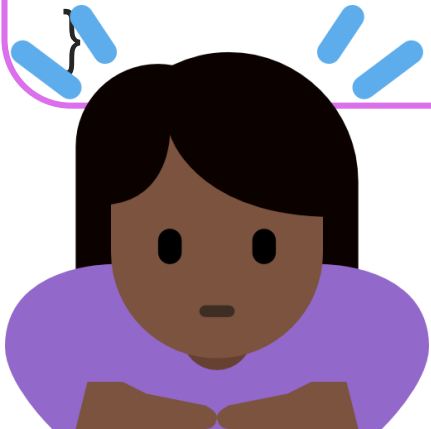
77

Since it is an art form, artists differ on what is more pretty

```
int main(){
    int a; // My first int
    int b; // The other int
    // Assign them values
    a = 5, b = 4;
    a + b;
    return 0;
```

```
int main(){
    int a; /* My first int */
    int b; /* The other int */
    /* Assign them values */
    a = 5, b = 4;
    a + b;
    return 0;
}
```

```
int main(){
    int a; // My first int
    int b; // The other int
    /* Assign them values */
    a = 5, b = 4;
    a + b;
    return 0;
}
```



Commenting Styles

77

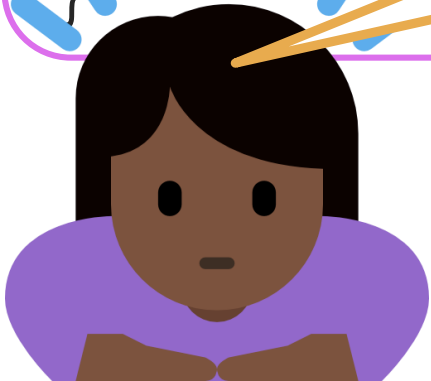
Since it is an art form, artists differ on what is more pretty

```
int main(){  
    int a; // My first int  
    int b; // The other int  
    // Assign them values  
    a = 5, b = 4;  
    a + b;  
    return 0;  
}
```

```
int main(){  
    int a; /* My first int */  
    int b; /* The other int */  
    /* Assign them values */  
    a = 5, b = 4;  
    a + b;  
    return 0;  
}
```

```
int main(){  
    int a; // My first int  
    int b; // The other int  
    /* Assign them values */  
    a = 5, b = 4;  
    a + b;  
    return 0;  
}
```

So I can mix
and match?



Commenting Styles

77

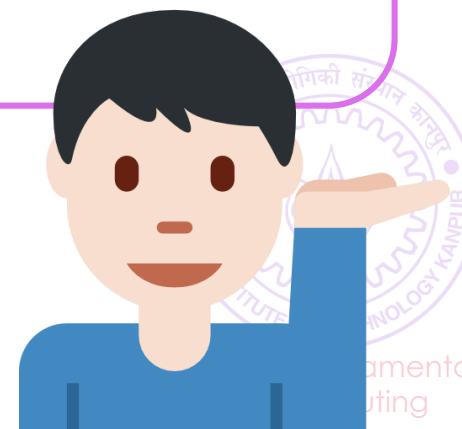
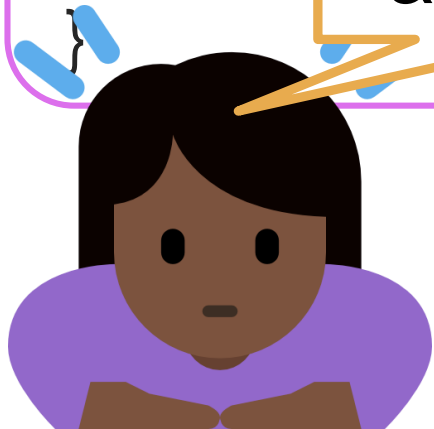
Since it is an art form, artists differ on what is more pretty

```
int main(){  
    int a; // My first int  
    int b; // The other int  
    // Assign them values  
    a = 5, b = 4;  
    a + b;  
    return 0;  
}
```

```
int main(){  
    int a; /* My first int */  
    int b; /* The other int */  
    /* Assign them values */  
    a = 5, b = 4;  
    a + b;  
    return 0;  
}
```

```
int main(){  
    int a; // My first int  
    int b; // The other int  
    /* Assign them values */  
    a = 5, b = 4;  
    a + b;  
    return 0;  
}
```

So I can mix
and match?



Commenting Styles

77

Since it is an art form, artists differ on what is more pretty

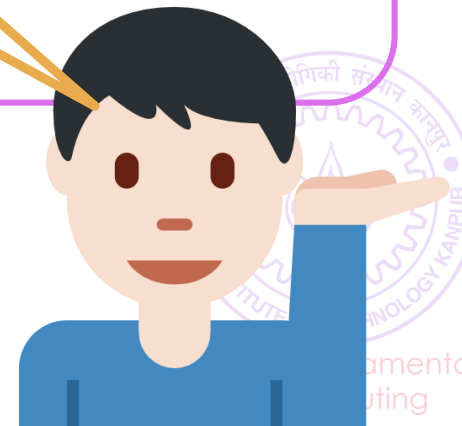
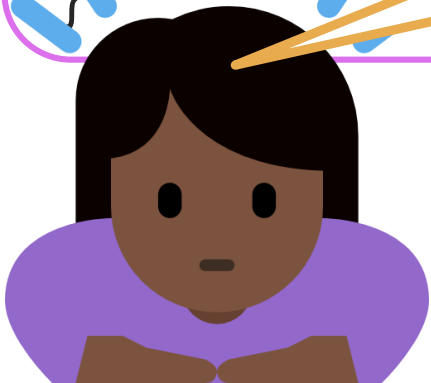
```
int main(){  
    int a; // My first int  
    int b; // The other int  
    // Assign them values  
    a = 5, b = 4;  
    a + b;  
    return 0;  
}
```

So I can mix
and match?

```
int main(){  
    int a; /* My first int */  
    int b; /* The other int */  
    /* Assign them values */  
    a = 5, b = 4;  
    a + b;  
    return 0;  
}
```

Yes. In fact /* */ is used
to comment several
lines at once – shortcut!

```
int main(){  
    int a; // My first int  
    int b; // The other int  
    /* Assign them values */  
    a = 5, b = 4;  
    a + b;  
    return 0;  
}
```



Commenting Styles

77

Since it is an art form, artists differ on what is more pretty

```
int main(){  
    int a; // My first int  
    int b; // The other int  
    // Assign them values  
    a = 5, b = 4;  
    a + b;  
    return 0;  
}
```

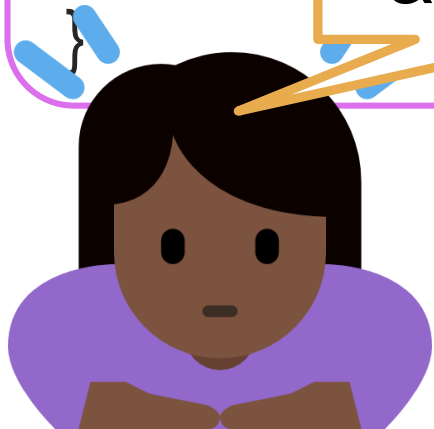
So I can mix
and match?

```
int main(){  
    int a; /* My first int */  
    int b; /* The other int */  
    /* Assign them values */  
    a = 5, b = 4;  
    a + b;  
    return 0;  
}
```

Yes. In fact /* */ is used
to comment several
lines at once – shortcut!

Just be a bit careful.
Some compilers don't
understand // comments

```
int main(){  
    int a; // My first int  
    int b; // The other int  
    /* Assign them values */  
    a = 5, b = 4;  
    a + b;  
    return 0;  
}
```



More on Comments

86



More on Comments

86

Use comments to describe why you defined each variable and what each step of your code is doing



More on Comments

86

Use comments to describe why you defined each variable and what each step of your code is doing

You will thank yourself for doing this when you are looking at your own code before the endsem exams 😊 😊



More on Comments

86

Use comments to describe why you defined each variable and what each step of your code is doing

You will thank yourself for doing this when you are looking at your own code before the endsem exams 😊 😊

Your team members in your company/research group will also thank you



More on Comments

86

Use comments to describe why you defined each variable and what each step of your code is doing

You will thank yourself for doing this when you are looking at your own code before the endsem exams 😊 😊

Your team members in your company/research group will also thank you

Multiline comments very handy. No need to write `//` on every line 😊



More on Comments

86

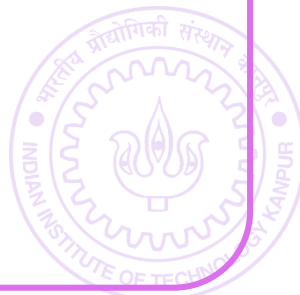
Use comments to describe why you defined each variable and what each step of your code is doing

You will thank yourself for doing this when you are looking at your own code before the endsem exams 😊 😊

Your team members in your company/research group will also thank you

Multiline comments very handy. No need to write `//` on every line 😊

```
int main(){  
    int a; // My first int  
    int b; // The other int  
    /* Assign them values  
       so that I can add  
       them later on */  
    a = 5, b = 4;  
    a + b;  
    return 0;  
}
```



More on Comments

86

Use comments to describe why you defined each variable and what each step of your code is doing

You will thank yourself for doing this when you are looking at your own code before the endsem exams 😊 😊

Your team members in your company/research group will also thank you

Multiline comments very handy. No need to write `//` on every line 😊

```
int main(){  
    int a; // My first int  
    int b; // The other int  
    // Assign them values  
    // so that I can add  
    // them later on  
    a = 5, b = 4;  
    a + b;  
    return 0;  
}
```

A handy tip while solving problems



A handy tip while solving problems

Comments can be also used to identify where is error



A handy tip while solving problems

Comments can be also used to identify where is error
Mr C will tell you (compile) where he thinks the error is



A handy tip while solving problems

Comments can be also used to identify where is error
Mr C will tell you (compile) where he thinks the error is
Commenting out lines can also help identify the error



A handy tip while solving problems 93

Comments can be also used to identify where is error
Mr C will tell you (compile) where he thinks the error is
Commenting out lines can also help identify the error

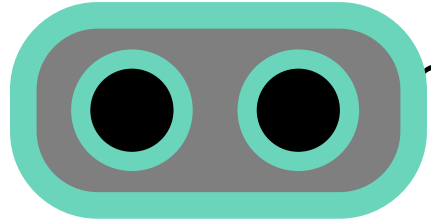
```
int main(){  
    int a, b;  
    c = a + b;  
    a = 5;  
    b = 4;  
    return 0;  
}
```



A handy tip while solving problems 93

Comments can be also used to identify where is error

Mr C will tell you (compile) where he thinks the error is



Printing out lines can also help identify the error

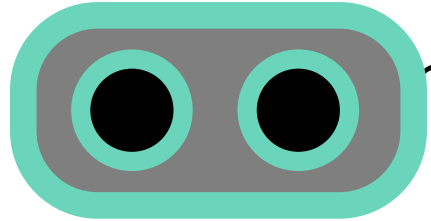
```
int main(){  
    int a, b;  
    c = a + b;  
    a = 5;  
    b = 4;  
    return 0;  
}
```



A handy tip while solving problems 93

Comments can be also used to identify where is error
Mr C will tell you (compile) where he thinks the error is

Error!



Printing out lines can also help identify the error

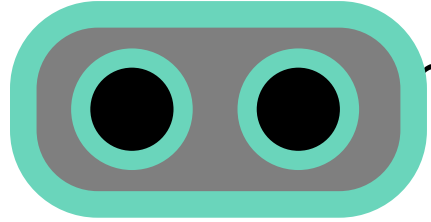
```
int main(){  
    int a, b;  
    c = a + b;  
    a = 5;  
    b = 4;  
    return 0;  
}
```



A handy tip while solving problems 93

Comments can be also used to identify where is error
Mr C will tell you (compile) where he thinks the error is

Error!



Printing out lines can also help identify the error

```
int main(){  
    int a, b;  
    c = a + b;  
    a = 5;  
    b = 4;  
    return 0;  
}
```

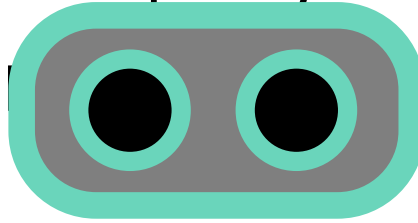
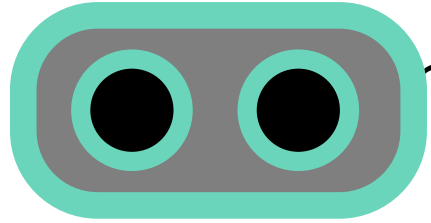
```
int main(){  
    int a, b;  
    // c = a + b;  
    a = 5;  
    b = 4;  
    return 0;  
}
```



A handy tip while solving problems 93

Comments can be also used to identify where is error
Mr C will tell you (compile) where he thinks the error is

Error!



```
int main(){  
    int a, b;  
    c = a + b;  
    a = 5;  
    b = 4;  
    return 0;  
}
```

```
int main(){  
    int a, b;  
    // c = a + b;  
    a = 5;  
    b = 4;  
    return 0;  
}
```

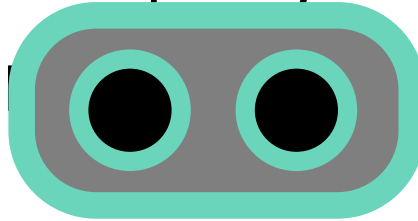
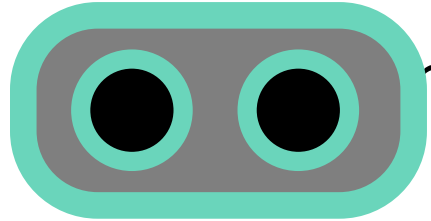


A handy tip while solving problems 93

Comments can be also used to identify where is error
Mr C will tell you (compile) where the error is

Error!

Okay!



```
int main(){  
    int a, b;  
    c = a + b;  
    a = 5;  
    b = 4;  
    return 0;  
}
```

```
int main(){  
    int a, b;  
    // c = a + b;  
    a = 5;  
    b = 4;  
    return 0;  
}
```

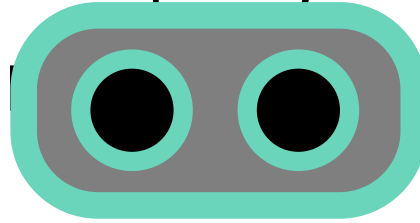
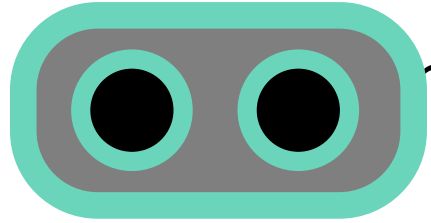


A handy tip while solving problems 93

Comments can be also used to identify where is error
Mr C will tell you (compile) where he thinks the error is

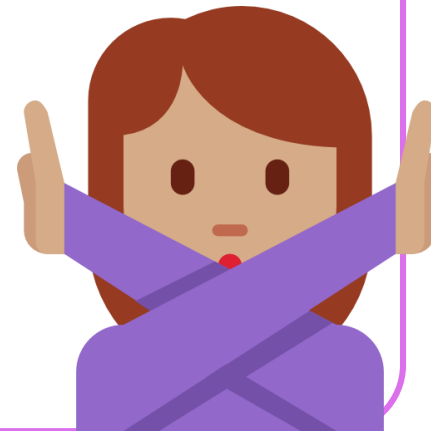
Error!

Okay!



```
int main(){  
    int a, b;  
    c = a + b;  
    a = 5;  
    b = 4;  
    return 0;  
}
```

```
int main(){  
    int a, b;  
    // c = a + b;  
    a = 5;  
    b = 4;  
    return 0;  
}
```

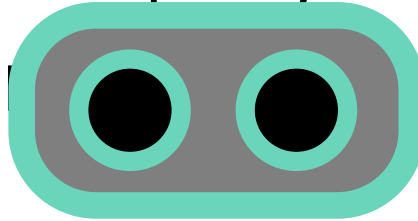
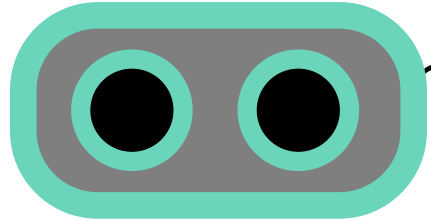


A handy tip while solving problems 93

Comments can be also used to identify where is error
Mr C will tell you (compile) where the error is

Error!

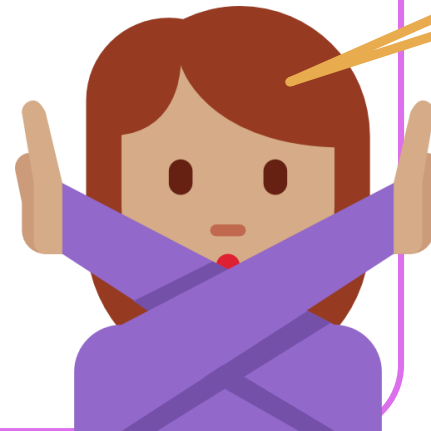
Okay!



```
int main(){  
    int a, b;  
    c = a + b;  
    a = 5;  
    b = 4;  
    return 0;  
}
```

```
int main(){  
    int a, b;  
    // c = a + b;  
    a = 5;  
    b = 4;  
    return 0;  
}
```

Aha! I forgot
to declare c

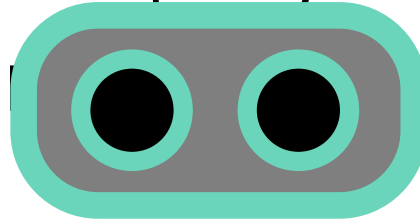
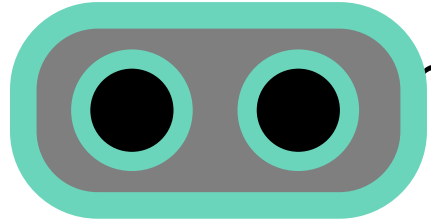


A handy tip while solving problems 93

Comments can be also used to identify where is error
Mr C will tell you (compile) where he thinks the error is

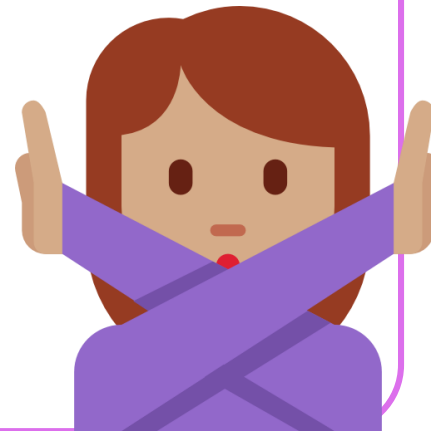
Error!

Okay!



```
int main(){  
    int a, b;  
    c = a + b;  
    a = 5;  
    b = 4;  
    return 0;  
}
```

```
int main(){  
    int a, b;  
    // c = a + b;  
    a = 5;  
    b = 4;  
    return 0;  
}
```

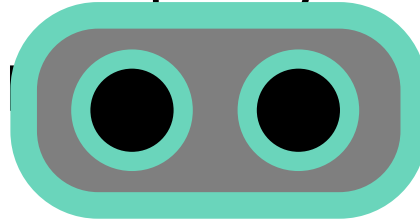
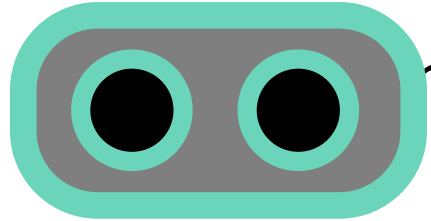


A handy tip while solving problems 93

Comments can be also used to identify where is error
Mr C will tell you (compile) where he thinks the error is

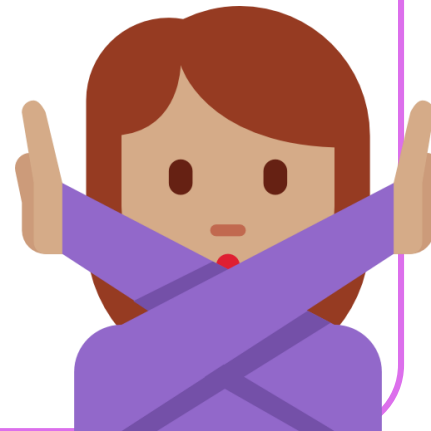
Error!

Okay!



```
int main(){  
    int a, b;  
    c = a + b;  
    a = 5;  
    b = 4;  
    return 0;  
}
```

```
int main(){  
    int a, b;  
    // c = a + b;  
    a = 5;  
    b = 4;  
    return 0;  
}
```



```
int main(){  
    int a, b, c;  
    c = a + b;  
    a = 5;  
    b = 4;  
    return 0;  
}
```

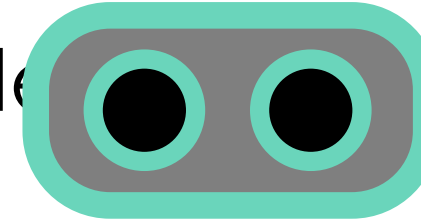
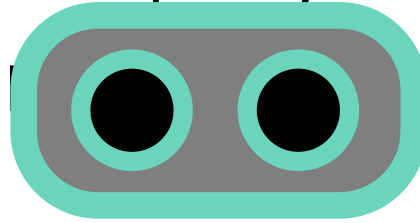
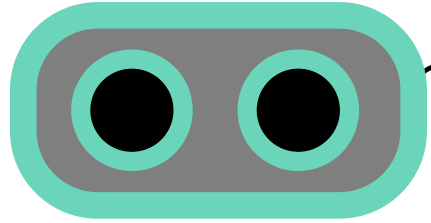


A handy tip while solving problems 93

Comments can be also used to identify where is error
Mr C will tell you (compile) where he thinks the error is
Printing out line numbers also help identify error

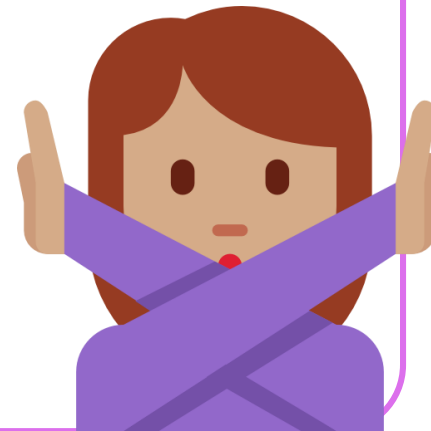
Error!

Okay!



```
int main(){  
    int a, b;  
    c = a + b;  
    a = 5;  
    b = 4;  
    return 0;  
}
```

```
int main(){  
    int a, b;  
    // c = a + b;  
    a = 5;  
    b = 4;  
    return 0;  
}
```



```
int main(){  
    int a, b, c;  
    c = a + b;  
    a = 5;  
    b = 4;  
    return 0;  
}
```



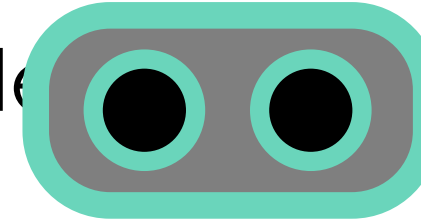
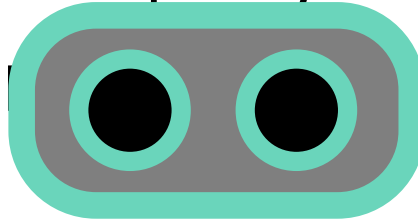
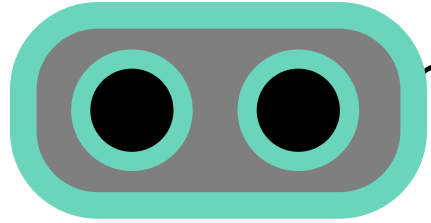
A handy tip while solving problems 93

Comments can be also used to identify where is error
Mr C will tell you (compile) where he thinks the error is

Error!

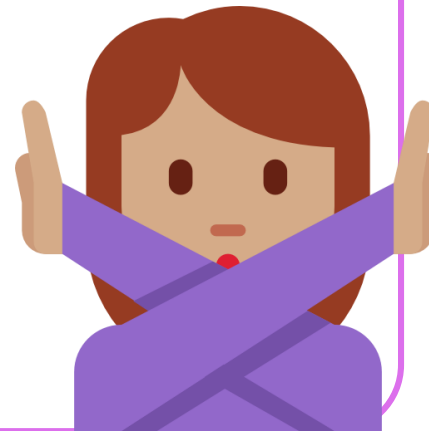
Okay!

Okay!



```
int main(){  
    int a, b;  
    c = a + b;  
    a = 5;  
    b = 4;  
    return 0;  
}
```

```
int main(){  
    int a, b;  
    // c = a + b;  
    a = 5;  
    b = 4;  
    return 0;  
}
```



```
int main(){  
    int a, b, c;  
    c = a + b;  
    a = 5;  
    b = 4;  
    return 0;  
}
```



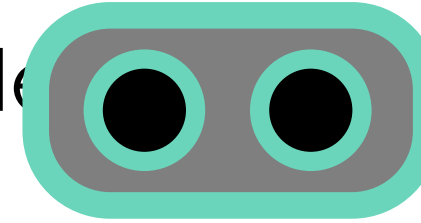
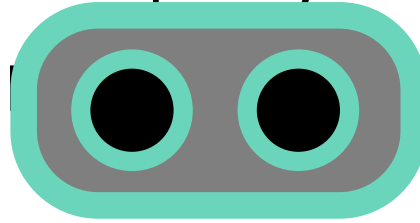
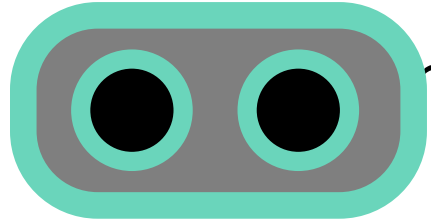
A handy tip while solving problems 93

Comments can be also used to identify where is error
Mr C will tell you (compile) where the error is

Error!

Okay!

Okay!



```
int main(){  
    int a, b;  
    c = a + b;  
    a = 5;  
    b = 4;  
    return 0;  
}
```

```
int main(){  
    int a, b;  
    // c = a + b;  
    a = 5;  
    b = 4;  
    return 0;  
}
```



```
int main(){  
    int a, b, c;  
    c = a + b;  
    a = 5;  
    b = 4;  
    return 0;  
}
```



Most Awaited scanf 😊



Most Awaited scanf 😊

- A bit frustrating



Most Awaited scanf 😊

- A bit frustrating
 - Mr C can add two numbers but both have to be written into code



Most Awaited scanf 😊

- A bit frustrating
 - Mr C can add two numbers but both have to be written into code
 - Also called “hardcoding”



Most Awaited scanf 😊

- A bit frustrating
 - Mr C can add two numbers but both have to be written into code
 - Also called “hardcoding”
 - A bit like a calculator which can only add 5 and 4 😊



Most Awaited scanf 😊

- A bit frustrating
 - Mr C can add two numbers but both have to be written into code
 - Also called “hardcoding”
 - A bit like a calculator which can only add 5 and 4 😊
 - To add 6 and 9, write a new calculator 😞



Most Awaited scanf 😊

- A bit frustrating
 - Mr C can add two numbers but both have to be written into code
 - Also called “hardcoding”
 - A bit like a calculator which can only add 5 and 4 😊
 - To add 6 and 9, write a new calculator 😞
- Can't we ask Mr C to request us for the numbers when he is executing our requests i.e. at *runtime*?



Most Awaited scanf 😊

- A bit frustrating
 - Mr C can add two numbers but both have to be written into code
 - Also called “hardcoding”
 - A bit like a calculator which can only add 5 and 4 😊
 - To add 6 and 9, write a new calculator 😞
- Can't we ask Mr C to request us for the numbers when he is executing our requests i.e. at *runtime*?
- Of course – take input from the user using scanf



Adding two unknown numbers 118



Adding two unknown numbers 118

HOW WE USUALLY SPEAK TO A HUMAN



Adding two unknown numbers 118

HOW WE USUALLY SPEAK TO A HUMAN



Adding two unknown numbers 118

HOW WE USUALLY SPEAK TO A HUMAN

Do you speak English?



Adding two unknown numbers 118

HOW WE USUALLY SPEAK TO A HUMAN

Do you speak English?
Hello



Adding two unknown numbers 118

HOW WE USUALLY SPEAK TO A HUMAN

Do you speak English?

Hello

a, b are two variables.



Adding two unknown numbers 118

HOW WE USUALLY SPEAK TO A HUMAN

Do you speak English?

Hello

a, b are two variables.

Please ask me for value of a.



Adding two unknown numbers 118

HOW WE USUALLY SPEAK TO A HUMAN

Do you speak English?

Hello

a, b are two variables.

Please ask me for value of a.

Please ask me for value of b.



Adding two unknown numbers 118

HOW WE USUALLY SPEAK TO A HUMAN

Do you speak English?

Hello

a, b are two variables.

Please ask me for value of a.

Please ask me for value of b.

Please print their sum.



Adding two unknown numbers 118

HOW WE USUALLY SPEAK TO A HUMAN

Do you speak English?

Hello

a, b are two variables.

Please ask me for value of a.

Please ask me for value of b.

Please print their sum.

Goodbye



Adding two unknown numbers 118

HOW WE MUST SPEAK TO MR. COMPILER

HOW WE USUALLY SPEAK TO A HUMAN

Do you speak English?

Hello

a, b are two variables.

Please ask me for value of a.

Please ask me for value of b.

Please print their sum.

Goodbye



Adding two unknown numbers 118

HOW WE MUST SPEAK TO MR. COMPILER

HOW WE USUALLY SPEAK TO A HUMAN

Do you speak English?

Hello

a, b are two variables.

Please ask me for value of a.

Please ask me for value of b.

Please print their sum.

Goodbye



Adding two unknown numbers 118

HOW WE MUST SPEAK TO MR. COMPILER

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

HOW WE USUALLY SPEAK TO A HUMAN

Do you speak English?

Hello

a, b are two variables.

Please ask me for value of a.

Please ask me for value of b.

Please print their sum.

Goodbye



Adding two unknown numbers 118

HOW WE MUST SPEAK TO MR. COMPILER

```
#include <stdio.h>  
int main(){
```

HOW WE USUALLY SPEAK TO A HUMAN

Do you speak English?

Hello

a, b are two variables.

Please ask me for value of a.

Please ask me for value of b.

Please print their sum.

Goodbye



Adding two unknown numbers 118

HOW WE MUST SPEAK TO MR. COMPILER

```
#include <stdio.h>
int main(){
int a, b;
```

HOW WE USUALLY SPEAK TO A HUMAN

Do you speak English?
Hello
a, b are two variables.
Please ask me for value of a.
Please ask me for value of b.
Please print their sum.
Goodbye



Adding two unknown numbers 118

HOW WE MUST SPEAK TO MR. COMPILER

```
#include <stdio.h>
int main(){
int a, b;
scanf("%d", &a);
```

HOW WE USUALLY SPEAK TO A HUMAN

Do you speak English?
Hello
a, b are two variables.
Please ask me for value of a.
Please ask me for value of b.
Please print their sum.
Goodbye



Adding two unknown numbers 118

HOW WE MUST SPEAK TO MR. COMPILER

```
#include <stdio.h>
int main(){
int a, b;
scanf("%d", &a);
scanf("%d", &b);
```

HOW WE USUALLY SPEAK TO A HUMAN

Do you speak English?
Hello
a, b are two variables.
Please ask me for value of a.
Please ask me for value of b.
Please print their sum.
Goodbye



Adding two unknown numbers 118

HOW WE MUST SPEAK TO MR. COMPILER

```
#include <stdio.h>
int main(){
int a, b;
scanf("%d", &a);
scanf("%d", &b);
printf("%d", a + b);
```

HOW WE USUALLY SPEAK TO A HUMAN

Do you speak English?
Hello
a, b are two variables.
Please ask me for value of a.
Please ask me for value of b.
Please print their sum.
Goodbye



Adding two unknown numbers 118

HOW WE MUST SPEAK TO MR. COMPILER

```
#include <stdio.h>
int main(){
int a, b;
scanf("%d", &a);
scanf("%d", &b);
printf("%d", a + b);
return 0;
```

HOW WE USUALLY SPEAK TO A HUMAN

Do you speak English?
Hello
a, b are two variables.
Please ask me for value of a.
Please ask me for value of b.
Please print their sum.
Goodbye



Adding two unknown numbers 118

HOW WE MUST SPEAK TO MR. COMPILER

```
#include <stdio.h>
int main(){
int a, b;
scanf("%d", &a);
scanf("%d", &b);
printf("%d", a + b);
return 0;
}
```

HOW WE USUALLY SPEAK TO A HUMAN

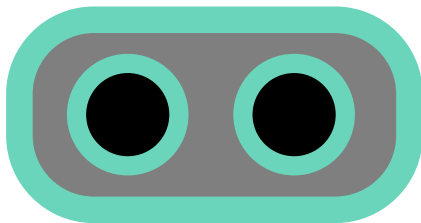
Do you speak English?
Hello
a, b are two variables.
Please ask me for value of a.
Please ask me for value of b.
Please print their sum.
Goodbye



Adding two unknown numbers 118

HOW WE MUST SPEAK TO MR. COMPILER

```
#include <stdio.h>
int main(){
int a, b;
scanf("%d", &a);
scanf("%d", &b);
printf("%d", a + b);
return 0;
}
```



HOW WE USUALLY SPEAK TO A HUMAN

Do you speak English?

Hello

a, b are two variables.

Please ask me for value of a.

Please ask me for value of b.

Please print their sum.

Goodbye



Adding two unknown numbers 118

HOW WE MUST SPEAK TO MR. COMPILER

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

```
int main(){
```

```
int a, b;
```

```
scanf("%d", &a);
```

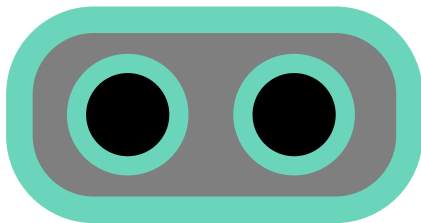
```
scanf("%d", &b);
```

```
printf("%d", a + b);
```

```
return 0;
```

```
}
```

a



HOW WE USUALLY SPEAK TO A HUMAN

Do you speak English?

Hello

a, b are two variables.

Please ask me for value of a.

Please ask me for value of b.

Please print their sum.

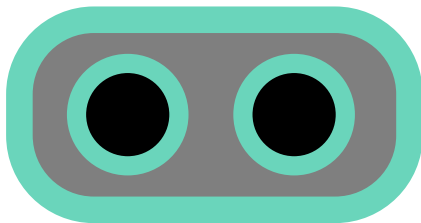
Goodbye



Adding two unknown numbers 118

HOW WE MUST SPEAK TO MR. COMPILER

```
#include <stdio.h>
int main(){
int a, b;
scanf("%d", &a);
scanf("%d", &b);
printf("%d", a + b);
return 0;
}
```



a

b

HOW WE USUALLY SPEAK TO A HUMAN

Do you speak English?

Hello

a, b are two variables.

Please ask me for value of a.

Please ask me for value of b.

Please print their sum.

Goodbye

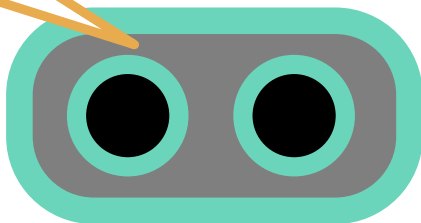


Adding two unknown numbers 118

HOW WE MUST SPEAK TO MR. COMPILER

```
#include <stdio.h>
int main(){
    int a, b;
    scanf("%d", &a);
    scanf("%d", &b);
    printf("%d", a + b);
    return 0;
}
```

Please give
me input



a

b

HOW WE USUALLY SPEAK TO A HUMAN

Do you speak English?

Hello

a, b are two variables.

Please ask me for value of a.

Please ask me for value of b.

Please print their sum.

Goodbye



Adding two unknown numbers 118

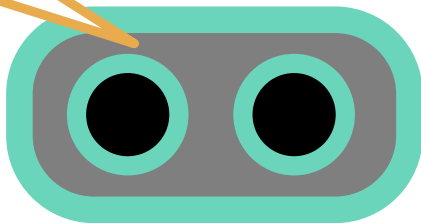
HOW WE MUST SPEAK TO MR. COMPILER

```
#include <stdio.h>
int main(){
    int a, b;
    scanf("%d", &a);
    scanf("%d", &b);
    printf("%d", a + b);
    return 0;
}
```

a

b

Please give
me input



HOW WE USUALLY SPEAK TO A HUMAN

Do you speak English?
Hello
a, b are two variables.
Please ask me for value of a.
Please ask me for value of b.

Console

Activity Log

Input

Output

Adding two unknown numbers 118

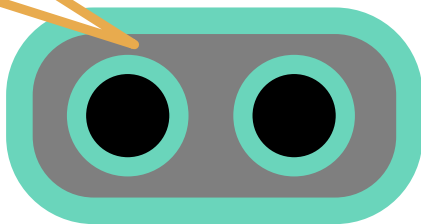
HOW WE MUST SPEAK TO MR. COMPILER

```
#include <stdio.h>
int main(){
    int a, b;
    scanf("%d", &a);
    scanf("%d", &b);
    printf("%d", a + b);
    return 0;
}
```

a

b

Please give
me input



HOW WE USUALLY SPEAK TO A HUMAN

Do you speak English?
Hello
a, b are two variables.
Please ask me for value of a.
Please ask me for value of b.

Console

Activity Log

Input

Output

3 8

Adding two unknown numbers 118

HOW WE MUST SPEAK TO MR. COMPILER

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

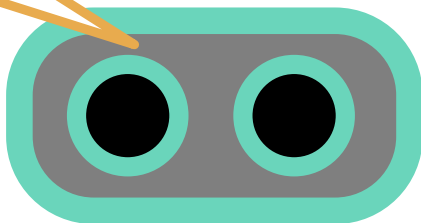
```
int main(){
```

Thanks. Let me
get back to work

Please give
me input

```
return 0;
```

```
}
```



b

HOW WE USUALLY SPEAK TO A HUMAN

Do you speak English?

Hello

a, b are two variables.

Please ask me for value of a.

Please ask me for value of b.

Console

Activity Log

Input

Output

38

Adding two unknown numbers 118

HOW WE MUST SPEAK TO MR. COMPILER

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

```
int main(){
```

```
int a, b;
```

```
scanf("%d", &a);
```

```
scanf("%d", &b);
```

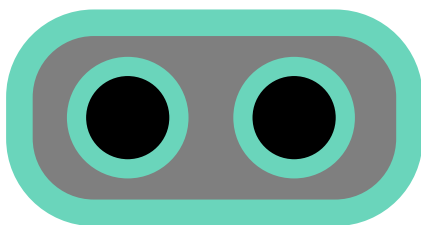
```
printf("%d", a + b);
```

```
return 0;
```

```
}
```

a

b



HOW WE USUALLY SPEAK TO A HUMAN

Do you speak English?

Hello

a, b are two variables.

Please ask me for value of a.

Please ask me for value of b.

Console

Activity Log

Input

Output

38

Adding two unknown numbers 118

HOW WE MUST SPEAK TO MR. COMPILER

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

```
int main(){
```

```
int a, b;
```

```
scanf("%d", &a);
```

```
scanf("%d", &b);
```

```
printf("%d", a + b);
```

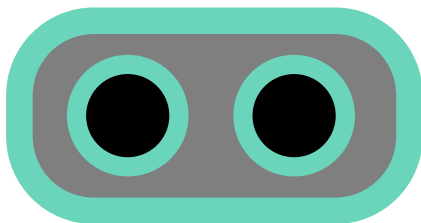
```
return 0;
```

```
}
```

3

a

b



HOW WE USUALLY SPEAK TO A HUMAN

Do you speak English?

Hello

a, b are two variables.

Please ask me for value of a.

Please ask me for value of b.

Console

Activity Log

Input

Output

3 8

Adding two unknown numbers 118

HOW WE MUST SPEAK TO MR. COMPILER

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

```
int main(){
```

```
int a, b;
```

```
scanf("%d", &a);
```

```
scanf("%d", &b);
```

```
printf("%d", a + b);
```

```
return 0;
```

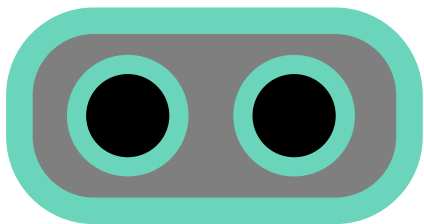
```
}
```

3

a

8

b



HOW WE USUALLY SPEAK TO A HUMAN

Do you speak English?

Hello

a, b are two variables.

Please ask me for value of a.

Please ask me for value of b.

Console

Activity Log

Input

Output

3 8

Adding two unknown numbers 118

HOW WE MUST SPEAK TO MR. COMPILER

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

```
int main(){
```

```
int a, b;
```

```
scanf("%d", &a);
```

```
scanf("%d", &b);
```

```
printf("%d", a + b);
```

```
return 0;
```

```
}
```

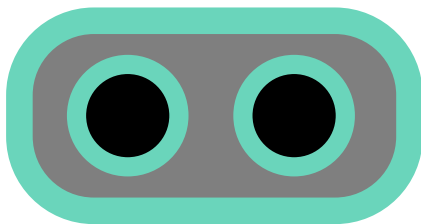
3

a

8

b

11



HOW WE USUALLY SPEAK TO A HUMAN

Do you speak English?

Hello

a, b are two variables.

Please ask me for value of a.

Please ask me for value of b.

Console

Activity Log

Input

Output

3 8

Adding two unknown numbers 118

HOW WE MUST SPEAK TO MR. COMPILER

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

```
int main(){
```

```
int a, b;
```

```
scanf("%d", &a);
```

```
scanf("%d", &b);
```

```
printf("%d", a + b);
```

```
return 0;
```

```
}
```

3

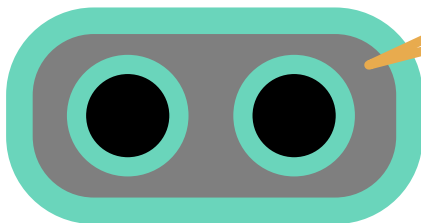
8

a

b

11

11



HOW WE USUALLY SPEAK TO A HUMAN

Do you speak English?

Hello

a, b are two variables.

Please ask me for value of a.

Please ask me for value of b.

Console

Activity Log

Input

Output

3 8

Words of Caution

150



Words of Caution

150

In Prutor, input has to be specified before “Execute”



Words of Caution

150

In Prutor, input has to be specified before “Execute”
Please be very careful about this common mistake



Words of Caution

150

In Prutor, input has to be specified before “Execute”
Please be very careful about this common mistake

```
scanf(“%d”,a);
```



Words of Caution

150

In Prutor, input has to be specified before “Execute”
Please be very careful about this common mistake

```
scanf(“%d”,a);
```



Words of Caution

150

In Prutor, input has to be specified before “Execute”
Please be very careful about this common mistake

```
scanf(“%d”,a);
```



```
scanf(“%d”,&a);
```



Words of Caution

150

In Prutor, input has to be specified before “Execute”
Please be very careful about this common mistake

`scanf(“%d”,a);`



`scanf(“%d",&a);`




Words of Caution

150

In Prutor, input has to be specified before “Execute”

Please be very careful about this common mistake

`scanf(“%d”,a);`  `scanf(“%d",&a);` 

Will explain what this & means, in few weeks



Words of Caution

150

In Prutor, input has to be specified before “Execute”
Please be very careful about this common mistake

`scanf(“%d”,a);`



`scanf(“%d”,&a);`



Will explain what this & means, in few weeks
Tomorrow, will learn shorthand for multiple inputs



Words of Caution

150

In Prutor, input has to be specified before “Execute”
Please be very careful about this common mistake

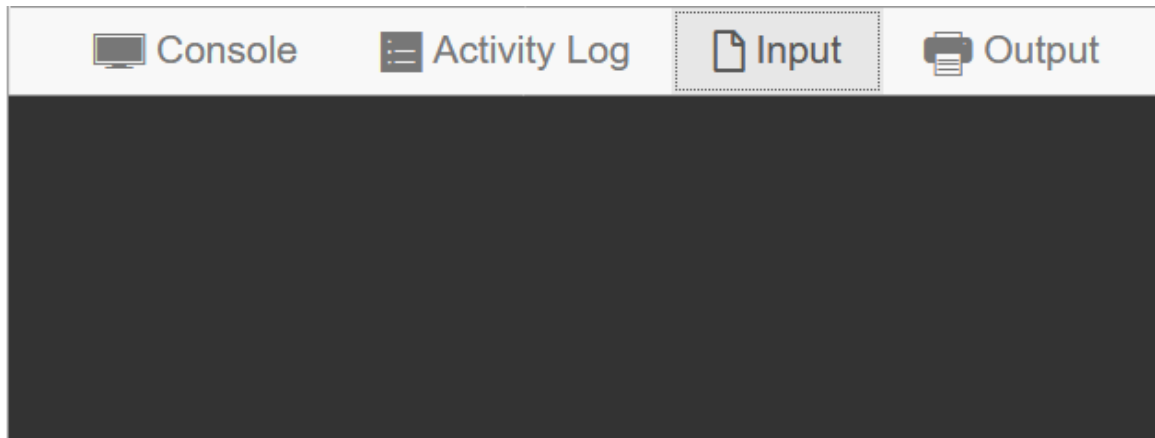
`scanf(“%d”,a);`



`scanf(“%d",&a);`



Will explain what this & means, in few weeks
Tomorrow, will learn shorthand for multiple inputs



Words of Caution

150

In Prutor, input has to be specified before “Execute”
Please be very careful about this common mistake

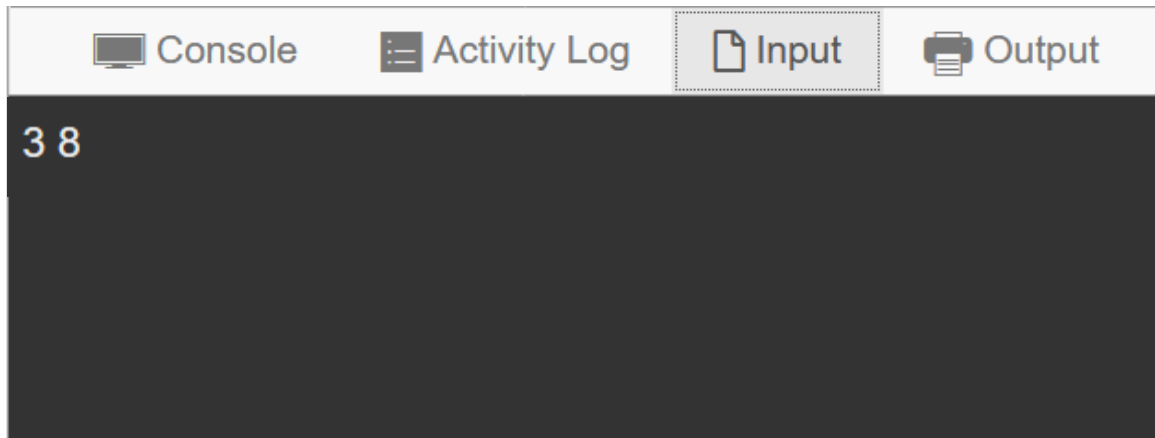
```
scanf(“%d”,a);
```



```
scanf(“%d",&a);
```



Will explain what this & means, in few weeks
Tomorrow, will learn shorthand for multiple inputs



Words of Caution

150

In Prutor, input has to be specified before “Execute”
Please be very careful about this common mistake

```
scanf(“%d”,a);
```

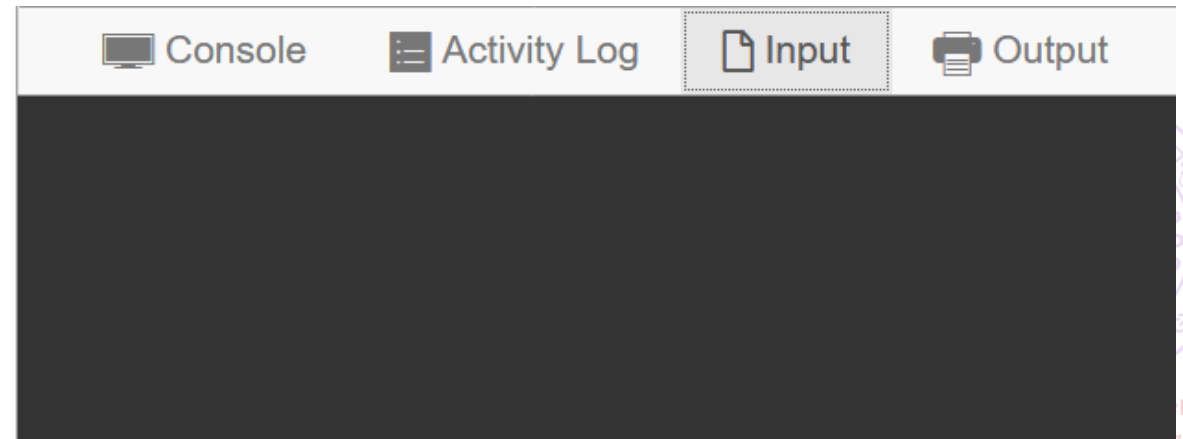
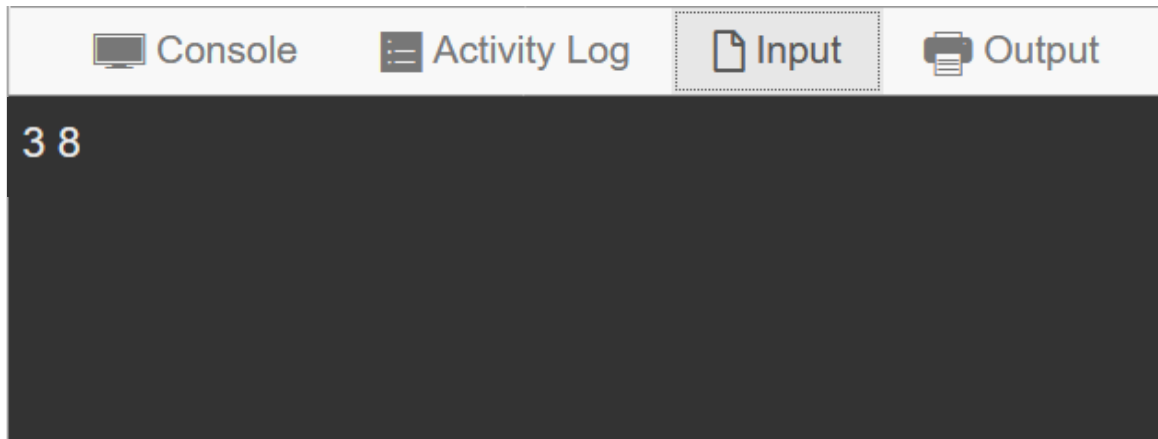


```
scanf(“%d”,&a);
```



Will explain what this & means, in few weeks

Tomorrow, will learn shorthand for multiple inputs



Words of Caution

150

In Prutor, input has to be specified before “Execute”
Please be very careful about this common mistake

```
scanf(“%d”,a);
```

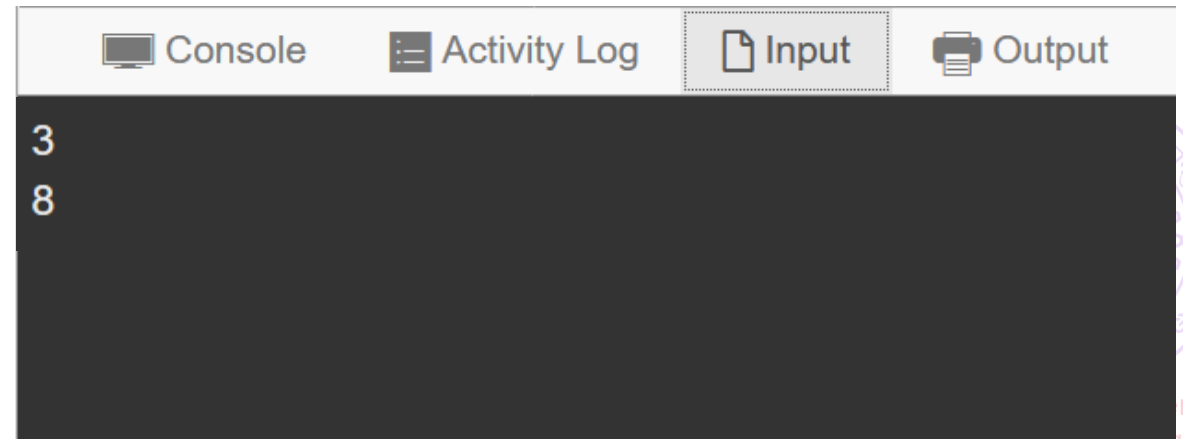
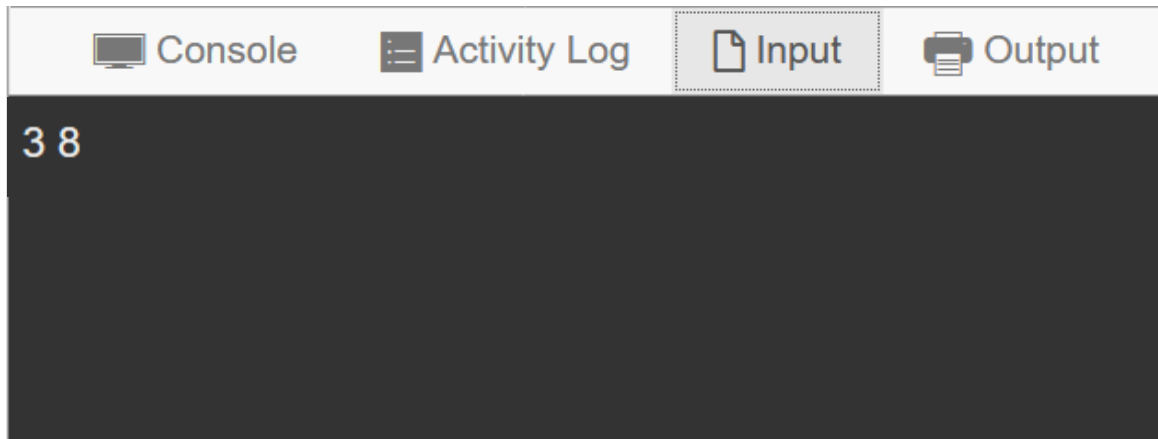


```
scanf(“%d",&a);
```



Will explain what this & means, in few weeks

Tomorrow, will learn shorthand for multiple inputs



Words of Caution

150

In Prutor, input has to be specified before “Execute”
Please be very careful about this common mistake

```
scanf(“%d”,a);
```



```
scanf(“%d",&a);
```



Will explain what this & means, in few weeks
Tomorrow, will learn shorthand for multiple input

Both work!
Experiment!

