```python
# Find  outputs  (Home  work)
a = 25  # Ref  'a'  points  to  object  25
b = 25   # Ref  'b'   points  to  same  object   25  becoz  int  is  immutable
and  reusable
print(a  is  b)   #   True  becoz  'a'  and  'b' point  to same  object  25
c = 'Hyd'   # Ref  'c'  points  to  object   'Hyd'
d = 'Hyd'  # Ref  'd'   points  to  same  object   'Hyd'   becoz   str  is
immutable  and  reusable
print(c  is  d)   #   True  becoz  'c'  and  'd' point  to same  object   'Hyd'
e = False   # Ref  'e'  points  to  object   False
f = False  # Ref  'f'   points  to  same  object   False   becoz   bool  is
immutable  and  reusable
print(e  is  f)   #   True  becoz  'e'  and  'f'  point  to same  object   False
g = range(10)   # Ref  'g'  points  to   range  object
h = range(10)  # Ref  'h'   points  to   another  range   object   becoz   range
is  immutable   but  not    reusable
print(g  is  h)   #  False  becoz  'g'  and  'h'  point  to   different  range
objects




'''
1) Can  there  be  multiple  int  objects  with  same  value ?  --->  No  becoz
int  object  is  immutable  and  reusable

2) Can  there  be  multiple  float  objects  with  same  value ?  --->  No  becoz
float  object  is  immutable  and  reusable

3) Can  there  be  multiple  string  objects  with  same  string ?  --->  No  becoz
str  object  is  immutable  and  reusable

4) Can  there  be  multiple  lists  with  same  elements ?  --->  Yes  becoz  list
is  mutable  and  not  reusable

5) Can  there  be  multiple  tuples  with  same  elements ?  ---> No  becoz  tuple
is  immutable  and  reusable

6) Can  there  be  multiple  sets  with  same  elements ?  --->  Yes  becoz  set
is  mutable  and  not  reusable

7) Can  there  be  multiple  dictionaries  with  same  key : value  pairs ?  --->
Yes  becoz  dict  is  mutable  and  not  reusable

8) Which  objects  are  reusable (5 + 2 = 7)  ?  --->  int , float , bool , complex
, NoneType , str   and   tuple

9) Which  objects  are  not  reusable (1 + 3 = 4) ?  --->  range , list , set ,
dict , bytearray

10) What  does  'is'  operator  do ?  --->  Compares  refernces  but  not  objects
     What  does  ==  operator  do ?  --->  Compares  objects
'''
```