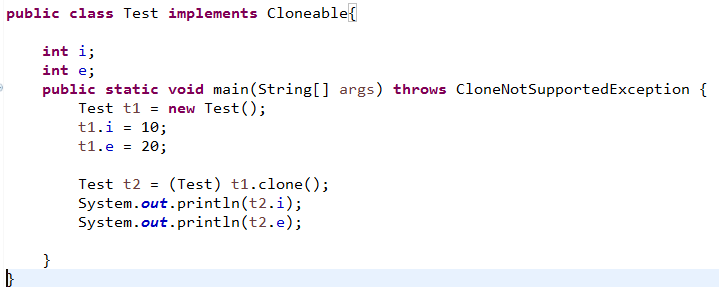
clone

1. Cloning: The process of creating exact duplicate object is **called cloning**.
2. **Need**:
   1. To maintain backup copy.
   2. If we make changes to object and later we want to know which changes have been made. So to preserve the state of object.
3. **How to clone**?
   1. By using clone() of Object class.
   2. 
4. Example:   
   

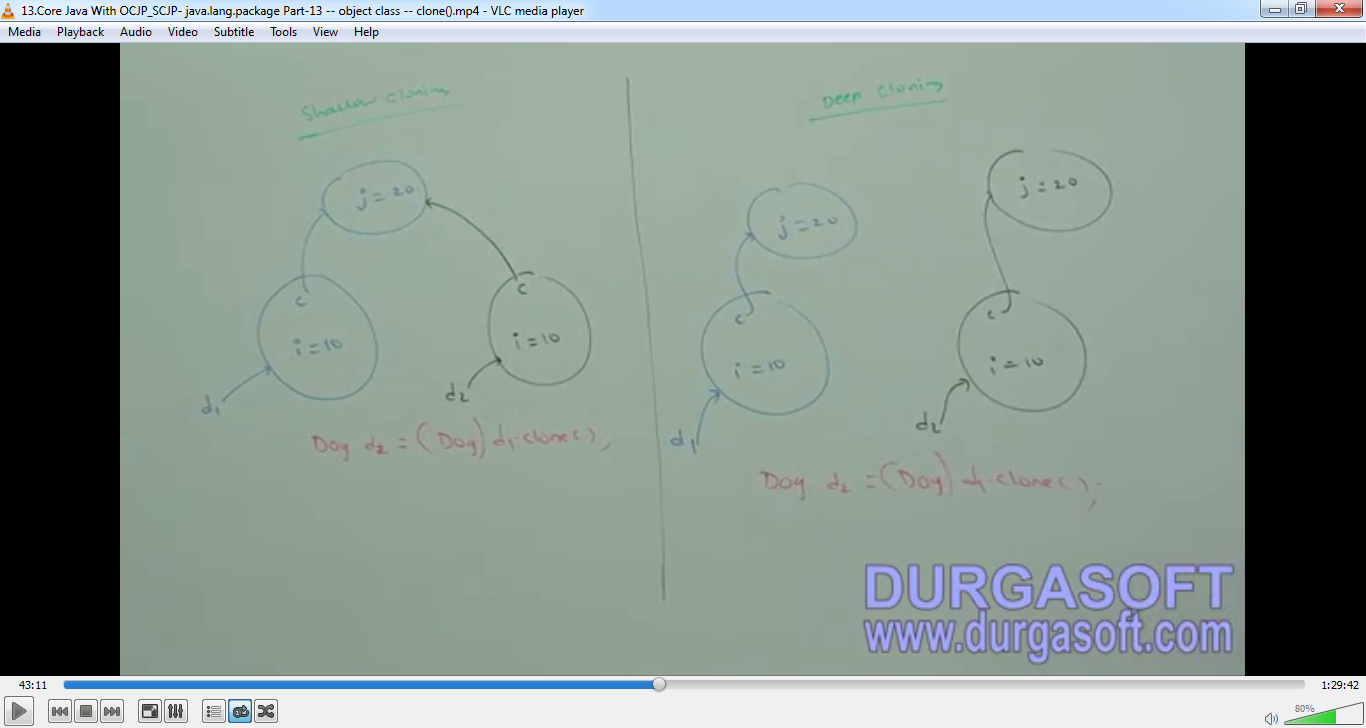
must implement Cloneable interface Marker  
Otherwise Runtime Exception.

Must handle or throws this Exception

Must typecase into appropriate type

1. d

Shallow vs Deep Cloning

1. 
2. **Shallow Cloning**:
   1. The process of creating bit-wise copy of an object. If the main object contains primitive instance variables then exactly duplicate copy will be created in the clone object. If the main object contains any reference variable then corresponding object will not be created just duplicate reference variable will be created pointing to old contained object.  
      **\*NOTE**: Object class clone() is meant for **shallow cloning**.
   2. Best if main object contains only primitive instance variables then only shallow cloning is perfect.
3. **Deep Cloning**:
   1. In deep cloning, if the main object contains any primitive instance variables then in the cloned object duplicate copy will be created. If the main object contains any reference variable then the corresponding contained object will also be created in the cloned copy.
   2. By default Object class clone() is meant for shallow cloning. But we can implement deep explicitly by overriding clone() in our class.
   3. **\*\*\*NOTE**: Property is responsible to implement deep copying   
      Example:   
      