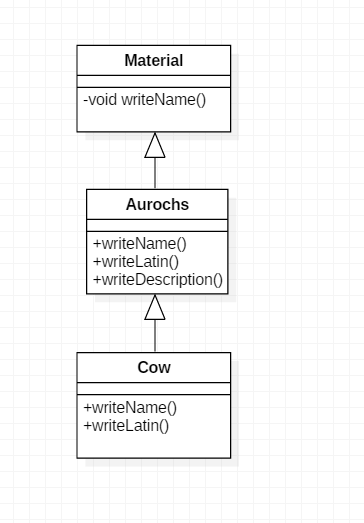
)a

Using Uml class diagram will show the relationship between the three classes. We can see from the following diagram that Aurochs inherits from Material and Cow inherits from Aurochs.

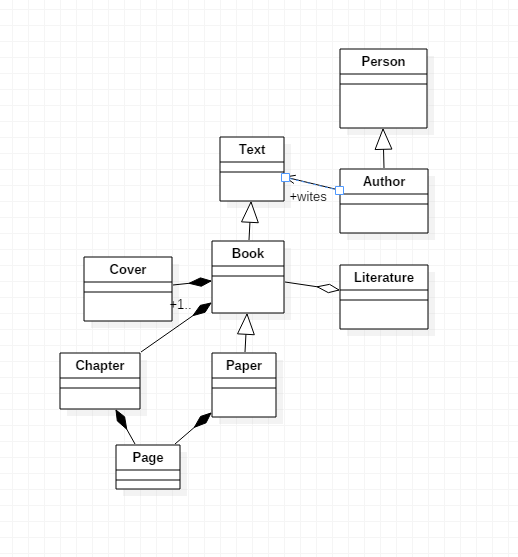
We also notice that each of the subclasses redefine writeName in its own implementation.



b)

|  |  |
| --- | --- |
| Statement | Output or Description of Failure |
| cow.writeName(); | Cow |
| aurochs.writeName(); | Cow |
| ((Material)aurochs).writeName(); | Material |
| ((Aurochs)cow).writeName(); | Cow |
| ((Cow)aurochs).writeName(); | Cow |
| aurochs.writeLatin(); | Bos primigenius |
| auerochse.writeDescription(); | Compile Error  auerochse can’t be resolved |
| ((Aurochs)material).writeName(); | Runtime Error  Material cannot be cast to Aurochs at Material.main |

2)



Cover is not a specific type Book. Instead we use Composition. When the book class is destroyed then the cover is destroyed as well. Direct association between Book and chapter is wrong. It should be Composition association as a Book is composed of chapters. Same applies for chapter pages.

Paper is specialization of Book that doesn’t have Cover or Chapters. There is also Composition relationship between Page and Paper without any relation to chapters.

The inheritance relationship between Book and literature is wrong. The Book should not be a child of literature as the child class is a specific type of the parent class and Book is not a specific type of literature. Instead we used Aggregation .Literature is made up of one or more books. Books will remain so even when the Literature is dissolved.