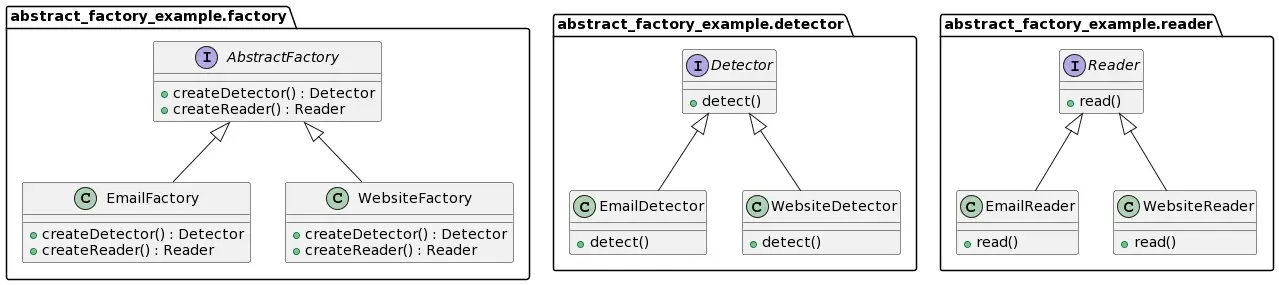
**AIM: Build Abstract Factory Design Pattern on the Model.**

Model chosen: Phishing Detection: Email and Website Reader and detector

**UML Diagram:**



1: UML Diagram

**CODE:**

|  |
| --- |
| interface Reader {      void read();  }  class EmailReader implements Reader {      public EmailReader() {          read();      }      public void read() {          System.out.println("Email Reader Object Created");      }  }  class WebsiteReader implements Reader {      public WebsiteReader() {          read();      }      public void read() {          System.out.println("Website Reader Object Created");      }  }  interface Detector {      void detect();  }  class EmailDetector implements Detector {      public EmailDetector() {          detect();      }      public void detect() {          System.out.println("Email Detector Object Created");      }  }  class WebsiteDetector implements Detector {      public WebsiteDetector() {          detect();      }      public void detect() {          System.out.println("Website Detector Object Created");      }  }  interface AbstractFactory {      Detector createDetector();      Reader createReader();  }  class EmailFactory implements AbstractFactory {      public Detector createDetector() {          return new EmailDetector();      }      public Reader createReader() {          return new EmailReader();      }  }  class WebsiteFactory implements AbstractFactory {      public Detector createDetector() {          return new WebsiteDetector();      }      public Reader createReader() {          return new WebsiteReader();      }  }  public class Test {      public static void main(String[] args) {          EmailFactory emailFactory = new EmailFactory();          emailFactory.createDetector();          emailFactory.createReader();          WebsiteFactory websiteFactory = new WebsiteFactory();          websiteFactory.createDetector();          websiteFactory.createReader();      }  } |

**Output:**

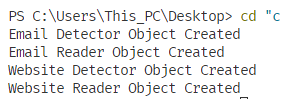


Figure 2: Output of the above program