|  |
| --- |
| **public** **class** Program  {  **public** **static** **void** main(String[] args)  {  Worker a = **new** Worker("Worker Tom", 5);  Supervisor b = **new** Supervisor("Supervisor Mary", 6);  Supervisor c = **new** Supervisor("Supervisor Jerry", 7);  Supervisor d = **new** Supervisor("Supervisor Bob", 9);  Worker e = **new** Worker("Worker Jimmy", 8);  //set up the relationships  b.addSubordinate(a); //Tom works for Mary  c.addSubordinate(b); //Mary works for Jerry  c.addSubordinate(d); //Bob works for Jerry  d.addSubordinate(e); //Jimmy works for Bob  //Jerry shows his happiness and asks everyone else to do the same  c.showHappiness();  }  }  **public** **interface** IEmployee  {  **void** showHappiness();  }  **public** **class** Worker **implements** IEmployee  {  **private** String name;  **private** **int** happiness;  **public** Worker(String name, **int** happiness)  {  **this**.name = name;  **this**.happiness = happiness;  }  **public** **void** showHappiness()  {  System.*out*.println(name + " showed happiness level of " + happiness);  }  }    **import** java.util.\*;    **public** **class** Supervisor **implements** IEmployee  {  **private** String name;  **private** **int** happiness;  **private** ArrayList<IEmployee> subordinate = **new** ArrayList<IEmployee>();  **public** Supervisor(String name, **int** happiness)  {  **this**.name = name;  **this**.happiness = happiness;  }  **public** **void** showHappiness()  {  System.*out*.println(name + " showed happiness level of " + happiness);  //show all the subordinate's happiness level  **for** (IEmployee i : subordinate)  i.showHappiness();  }  **public** **void** addSubordinate(IEmployee employee)  {  subordinate.add(employee);  }  } |

1. Draw UML class diagram?