**public** **class** WeatherRunner

{

**public** **static** **void** main(String[]args)

{

Weather su = **new** Sunny();

Weather st = **new** Stormy();

Weather sn = **new** Snowy();

su.display();

su.emitLight();

su.emitParticles();

su.emitTemperature();

su.setEmitTemperatureBehavior(**new** LowTemp());

su.emitTemperature();

System.***out***.println();

st.display();

st.emitLight();

st.emitParticles();

st.emitTemperature();

st.setEmitLightBehavior(**new** MaxLight());

st.emitLight();

System.***out***.println();

sn.display();

sn.emitLight();

sn.emitParticles();

sn.emitTemperature();

sn.setEmitParticlesBehavior(**new** NoRain());

sn.emitParticles();

}

}

**public** **abstract** **class** Weather

{

EmitParticlesBehavior emitParticlesBehavior;

EmitLightBehavior emitLightBehavior;

EmitTemperatureBehavior emitTemperatureBehavior;

**abstract** **void** display();

**public** **void** emitParticles()

{

emitParticlesBehavior.emitParticles();

}

**public** **void** emitLight()

{

emitLightBehavior.emitLight();

}

**public** **void** emitTemperature()

{

emitTemperatureBehavior.emitTemperature();

}

**public** **void** setEmitLightBehavior(EmitLightBehavior elb)

{

emitLightBehavior = elb;

}

**public** **void** setEmitParticlesBehavior(EmitParticlesBehavior epb)

{

emitParticlesBehavior = epb;

}

**public** **void** setEmitTemperatureBehavior(EmitTemperatureBehavior etb)

{

emitTemperatureBehavior = etb;

}

}

**public** **interface** EmitLightBehavior

{

**void** emitLight();

}

**public** **interface** EmitParticlesBehavior

{

**void** emitParticles();

}

**public** **interface** EmitTemperatureBehavior

{

**void** emitTemperature();

}

**public** **class** LowLight **implements** EmitLightBehavior

{

**public** **void** emitLight()

{

System.***out***.println("I'm emitting low light");

}

}

**public** **class** HalfLight **implements** EmitLightBehavior

{

**public** **void** emitLight()

{

System.***out***.println("I'm emitting half light");

}

}

**public** **class** MaxLight **implements** EmitLightBehavior

{

**public** **void** emitLight()

{

System.***out***.println("I'm emitting maximum light");

}

}

**public** **class** LowTemp **implements** EmitTemperatureBehavior

{

**public** **void** emitTemperature()

{

System.***out***.println("I'm emitting low temp");

}

}

**public** **class** MediumTemp **implements** EmitTemperatureBehavior

{

**public** **void** emitTemperature()

{

System.***out***.println("I'm emitting medium temp");

}

}

**public** **class** HighTemp **implements** EmitTemperatureBehavior

{

**public** **void** emitTemperature()

{

System.***out***.println("I'm emitting high temp");

}

}

**public** **class** NoRain **implements** EmitParticlesBehavior

{

**public** **void** emitParticles()

{

System.***out***.println("I'm not raining");

}

}

**public** **class** SomeRain **implements** EmitParticlesBehavior

{

**public** **void** emitParticles()

{

System.***out***.println("I'm raining a little");

}

}

**public** **class** Rain **implements** EmitParticlesBehavior

{

**public** **void** emitParticles()

{

System.***out***.println("I'm raining");

}

}

**public** **class** Sunny **extends** Weather

{

**public** Sunny()

{

emitLightBehavior = **new** MaxLight();

emitParticlesBehavior = **new** NoRain();

emitTemperatureBehavior = **new** HighTemp();

}

@Override

**void** display()

{

System.***out***.println("I'm Sunny weather");

}

}

**public** **class** Stormy **extends** Weather

{

**public** Stormy()

{

emitLightBehavior = **new** HalfLight();

emitParticlesBehavior = **new** Rain();

emitTemperatureBehavior = **new** MediumTemp();

}

**public** **void** display()

{

System.***out***.println("I'm Stormy Weather");

}

}

**public** **class** Snowy **extends** Weather

{

**public** Snowy()

{

emitLightBehavior = **new** LowLight();

emitParticlesBehavior = **new** SomeRain();

emitTemperatureBehavior = **new** LowTemp();

}

**public** **void** display()

{

System.***out***.println("I'm Snowy Weather");

}

}

Execution



