**Uppgift 1 - Perpetum Mobile**

**Deluppgift 1 — Fikapaus**

Kompilera programmet med javac BallWorld.java och provkör det med java BallWorld. Beskriv med några få ord vad som händer.

Svar:

Det som händer är att det startar upp en ruta där det är en gul boll som studsar fram och tillbaka mellan hörnen.

**Deluppgift 2 — Färgblindhet**

Byt färg på bollen (till t.ex. vit) utan att ändra i konstruktorn Ball() där standardfärgen sätts. Däremot behöver du göra ett anrop till en metod som sätter färgen för en boll. Vilket anrop och var i koden den skall utföras ska du lista ut själv. Beskriv med några få ord vilka ändringar du gjorde.

Svar:

Det fanns en setter metod för att ändra färgen så jag anropade den efter objektet ball var skapat.

ball.setColor(Color.white);

**Deluppgift 3 — Fängslande**

En "bounding box" är en rektangel som anger gränserna i vilken bollen studsar inom. Om du ändrar storleken på fönstret upptäcker du att området inom vilket bollen studsar är oförändrat. Metoden wasResized() anropas då fönsterstorleken ändras. Modifiera wasResized() så att "bounding boxen" ändras till dess rätta storlek. Titta på andra ställen i koden för att få en uppfattning vilken metod som avses och hur den ska användas. Beskriv vilka ändringar du gjorde.

Svar:

Jag uppdaterade bollens bounding-box med dom nya värdena för width och height.

ball.setBoundingBox( new Rectangle( 0, 0, newWidth, newHeight ) );

**Deluppgift 4 — Storlek**

I klassen Ball finns metoder för att ändra färg respektive "bounding box", men det finns ingen metod för att sätta bollens storlek (diameter). Tillfoga en sådan metod och använd den sedan för att ändra storleken på bollen. Beskriv vilka ändringar du gjorde.

Jag la till en setter för att kunna justera storleken.

public void setDiameter(int d) {

diameter = d;

}

Sedan anropade jag metoden där ballPanel initierar attributen för ett nytt Ball objekt.

ball.setDiameter(100)

**Deluppgift 5 — Dubbelt så många**

Lägg in en extra boll med annan färg, storlek och hastighet så att det finns totalt två bollar som studsar. Beskriv vilka ändringar du gjorde.

I klassen BallPanel så skapade jag ett nytt ball objekt ball2. Sedan anropade jag samma metoder som ball redan använt så som setcolor, setBoundingBox, paint, action osv.

**Deluppgift 6 — Pulsarer**

Låt bollarna omväxlande växa och krympa. För detta behöver du bestämma minimistorlek samt maximistorlek för bollar (dessa gränser väljs med förstånd). Du måste också lägga till en instansvariabel i klassen Ball som anger om bollen just nu krymper eller växer. Denna variabel ska vara av typen boolean. Vidare behöver du kod som växlar variabelns värde då bollen når sin minimistorlek och maximistorlek.

Modifiera metoden action() så att bollen får den pulserande egenskapen. Beskriv vilka ändringar du gjorde.

Förutom att initiera fältet growing på samma sätt som så la jag till följande kod. En metod som sätter om en boll växer eller inte. Den tar in storleken av en boll och returnerar falskt om den är för liten och sant om den är för stor. Sedan anropar jag metoden i action. Där ändrar jag på diametern om den är för stor eller ej.

public void setGrowingState(int diameter) {  
  
 if(diameter > 100) {  
 growing = false;  
 }  
 if(diameter <= 10){  
 growing = true;  
 }  
 }  
  
 // Flytta bollen med aktuell riktning och hastighet ett steg  
 public void action() {  
 x = x + dx;  
 y = y + dy;  
  
 //Uppgift 6  
  
 setGrowingState(diameter);  
 if (growing){  
 diameter ++;  
 }else{  
 diameter --;  
 }  
  
  
 constrain();  
 }  
}

Här är en alternativ lösning med två if satser.

Alternativ lösning:  
  
if(diameter >= 80) {  
 growing = false;  
}  
if(diameter <= 3){  
 growing = true;  
} if (growing){  
 diameter ++;  
}else{  
 diameter --;  
}