2016年度ネットワーク基礎　レポート１　テンプレート

学籍番号051650179　氏名　バラク・エリラム

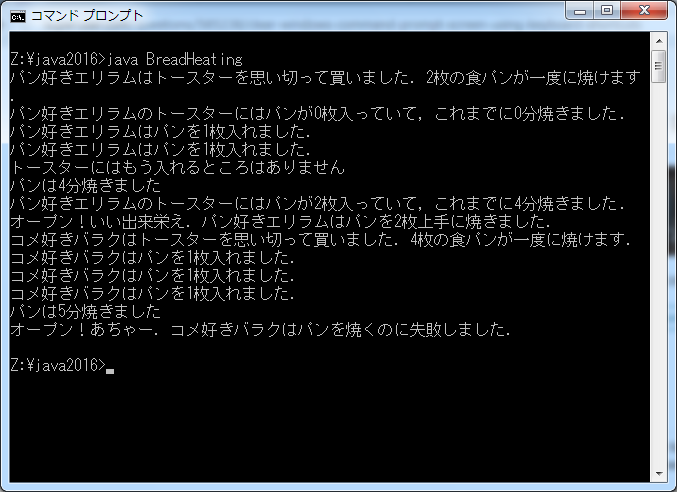
※各欄の高さや文字の大きさ，間隔などは適宜調整すること

１）

・ソースコードの空欄

|  |  |
| --- | --- |
| 1 | bread=0;  minutes=0;  max=m; |
| 2 | bread<max |
| 3 | "トースターにはもう入れるところはありません" |
| 4 | void heat(int h){  minutes=minutes+h;  System.out.println("パンは"+h+"分焼きました");  } |
| 5 | "オープン！" |
| 6 | minutes==(2\*bread) |
| 7 | heated=bread; |
| 8 | bread=0;  minutes=0; |
| 9 | eliram.addBread1(); |
| 10 | barak |
| 11 | "コメ好きバラク",4 |
| 12 | barak.heat(5); |

・実行結果のスナップショット（動作を示したコマンドプロンプトの画像）

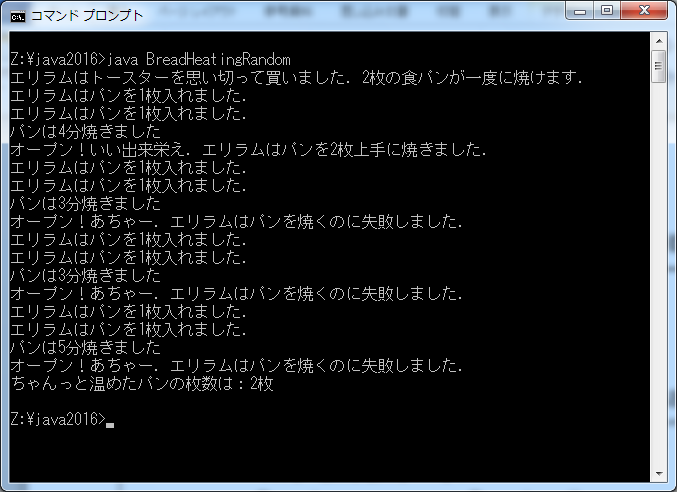


２）

・作成したBreadHeatingRandom.javaのソースコード

|  |
| --- |
| //BreadHeatingRandom.java  import java.util.\*;  public class BreadHeatingRandom{  public static void main(String args[]){  Toaster eliram=new Toaster("エリラム", 2);  int amount= 0;  Random r= new Random();  //ここにコードを追加  for(int i=0;i<4;i++){  eliram.addBread1();  eliram.addBread1();  eliram.heat(r.nextInt(3)+3);  amount=(amount+eliram.open());  }  System.out.println("ちゃんっと温めたパンの枚数は："+amount+"枚");  }  } |

・実行結果のスナップショット

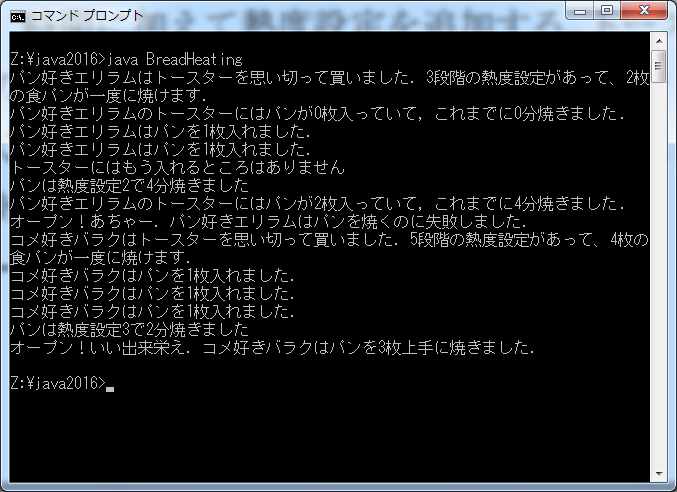


３）

・考えた機能とソースコードの追加・改変個所の説明

　時間に加えて熱度設定を追加する。トースタークラスに熱度設定（powerMax）と焼く効果（bake）のフィールドを追加し、コンストラクタでトースターにいくら熱度設定があるのかを定義する。焼く効果は熱度かける時間とみなす。前と同じパンをちゃんと焼くために、焼く効果は枚数の二倍である必要がある。

・実行結果のスナップショット



* 作成・改変したソースコード（以下に記載することに加え，各ファイル自体もWebから提出すること．）

改変したToaster.java（改変箇所がわかるように色を変えること）

|  |
| --- |
| //Toaster.java  class Toaster{  String owner;  int bread;  int minutes;  int max;  int powerMax;  int bake;  Toaster(String o, int m,int d) {  owner= o;  bread=0;  minutes=0;  max=m;  powerMax=d;  bake=0;  System.out.println(owner+"はトースターを思い切って買いました．"+powerMax+"段階の熱度設定があって、"+max+"枚の食パンが一度に焼けます．");  }    void addBread1(){  if(bread<max){  bread++;  System.out.println(owner+"はパンを1枚入れました．");  }else{  System.out.println("トースターにはもう入れるところはありません");  }  }    void heat(int h,int p){  if(0<p&&p<=powerMax){  minutes=minutes+h;  bake=bake+h\*p;  System.out.println("パンは熱度設定"+p+"で"+h+"分焼きました");  }  else System.out.println("トースターにこの熱度設定はありません");  }    int open(){  int heated= 0;  //"println"でなくて"print"なのに注意．printの場合は最後の改行なし．  System.out.print("オープン！");  if(bake==(2\*bread)){  System.out.println("いい出来栄え．"+owner+"はパンを"  +bread+"枚上手に焼きました．");  heated=bread;  }else{  System.out.println("あちゃー．"+owner+"はパンを焼くのに失敗しました．");  }  bread=0;  minutes=0;  　　　　bake=0;  return(heated);  }    void printStatus(){  System.out.println(owner+"のトースターにはパンが"+bread  +"枚入っていて，これまでに"+minutes+"分焼きました.");  }  } |

改変したBreadHeating.java（改変箇所がわかるように色を変えること）

|  |
| --- |
| //BreadHeating.java  public class BreadHeating{  public static void main(String args[]){  Toaster eliram=new Toaster("パン好きエリラム", 2,3);  eliram.printStatus();  eliram.addBread1();  eliram.addBread1();  eliram.addBread1();  eliram.heat(4,5);  eliram.heat(4,2);  eliram.printStatus();  eliram.open();    Toaster barak= new Toaster("コメ好きバラク",4,5);    for(int i=0;i<3;i++){  barak.addBread1();  };  barak.heat(2,3);  barak.open();    }  } |

・実行結果のスナップショット

