Projeto Puzzle ,UFCD: Integração de Sistemas de Informação - Ferramentas

Curso: TPSI 1015ª,Formador : Luis Filipe

A UFCD vai ser avaliada por dois projetos 50% cada um , o

1º Trabalho prático e de avaliação da ufcd criar uma aplicação para Windows / Android.

Conteúdos abordados: JavaFx, Swing, Ficheiros, Classes, Dados Dinâmicos, Timer.

O Jogo dispõe de três variantes Letras, Romanos, Binário.

Deteta o fim do jogo, captura os dados jogados tempo de jogo, lista ao desempenhos dos diferentes jogados dados guardados num ficheiro, passados para dados dinâmicos arraylist e listados numa tabela.

Duas classes auxiliares uma para as células pertencentes ao jogo e outra referente aos dados dos diferentes jogadores e seus tempos.

A aplicação utiliza layot gridLayout, e JButton onde se iram ser despoletados eventos.

Dispõe de um Menu onde se pode ser alterado as diferentes variantes de jogo, o nome do jogador e uma opção automática de como resolver o jogo e as outras opções de Menu.

Aspeto da App.

|  |  |
| --- | --- |
|  |  |
|  |  |

Classe Célula:

**int Pos;**

**String Num;**

**boolean Mov;**

**boolean Hole;**

Class Jogador:

**String Nome;**

**String Data;**

**Float tempo;**

Outros dados:

Date data1,

GregorianCalendar calendar1.

List <Celula> quadro = new ArrayList <Celula>();

public List <Celula> CarregarDados (Object [][] ob){

date1 = new Date();

calendar1 = GregorianCalendar.getInstance();

calendar1.setTime(date1);

Horas1 = calendar1.get(Calendar.HOUR);

Minutos1 = calendar1.get(Calendar.MINUTE);

Segundos1 = calendar1.get(Calendar.SECOND);

Exemplo de gravar dados e importar dados

|  |  |
| --- | --- |
| public static void importar(List<Task> ls){  //dados locais  Object [] dados = new Object[3];  StringTokenizer leitura ;  Task tar;  int ct=0;  try{  FileReader arq = new FileReader("agenda.txt");  BufferedReader lerArq = new BufferedReader(arq);  String linha = lerArq.readLine();  // lê a primeira linha  // a variável "linha" recebe o valor "null" quando o processo  // de repetição atingir o final do arquivo texto  while (linha != null) {  // System.out.printf("\n\t%s\n",linha);  //para decompor a linha tem tres dados  //tarefa, prioridade e concluido  leitura = new StringTokenizer(linha);  ct = 0;  while(leitura.hasMoreElements()){  dados[ct] = leitura.nextElement();  ct++; }  tar = new Task((String) dados[0],  Integer.parseInt((String) dados[1]),  Boolean.parseBoolean((String) dados[2]));  ls.add(tar);  // lê da segunda até a última linha  linha = lerArq.readLine();  }//while  arq.close();  } catch (IOException e) {  System.err.printf("Erro na abertura do arquivo: %s.", e.getMessage());}  System.out.println("\n");  }//end metodo | //\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  //Metodo  public static void exportar(List <Task> ls) throws IOException {  //dados locais  FileWriter arq = new FileWriter("agenda.txt");  PrintWriter gravarArq = new PrintWriter(arq);  int i, n = ls.size();  for (i=0; i<n; i++) {  gravarArq.printf("%25s\t", ls.get(i).getTarefa());  gravarArq.printf("%5d\t", ls.get(i).getPrioridade());  gravarArq.printf("%8b\n", ls.get(i).getConcluido());  }  gravarArq.close();  System.out.println("\n");  }//end metodo |