Aula 07

Genéricos e Coleções

Genéricos

AMBIENT
INFORMÁTICA

Motivação para Genéricos:

- Bug é uma realidade na produção de software
- Bugs de compilação são mais fáceis de serem detectados que bugs de execução
- Generics faz com que alguns bugs, que eram detectados apenas em tempo de executação, sejam detectados em tempo de compilação

Classes genéricas

```
package aula07.genericos;
public class Caixa <E>{

    private E objeto;

    public void add(E objeto) {
        this.objeto = objeto;
    }

    public E get() {
        return objeto;
    }
}
```

Classes genéricas

```
package aula07.genericos;
public class CaixaDemo {
   public static void main(String[] args) {

      Caixa<Integer> caixaDeInteiros = new Caixa<Integer>();

      caixaDeInteiros.add(10);
      Integer someInteger = caixaDeInteiros.get();
      System.out.println(someInteger);
   }
}
```

Algumas observações:

- Uma classe genérica pode ser derivada de uma classe não genérica;
- Uma classe genérica pode ser derivada de uma classe genérica;
- Uma classe não genérica pode ser derivada de uma classe genérica;
- Um método genérico em uma subclasse pode sobrescrever um método genérico em uma superclasse se os dois métodos tiverem

Coleções

Visão geral

- Coleção é uma estrutura de dados;
- Armazena referencias a outros objetos;
- Antes do JSE5.0 as coleções manipulavam referencia à classe Object;
- No JSE5.0 a estrutura de coleções foi aprimorada com a capacidade de se especificar um tipo exato que será armazenado;
- Esta no pacote java.util

Tipos de coleções

- Collection
- Interface raiz de Set e List
- Set – não contém duplicatas
- List – pode conter elementos duplicados
- Map – associa chaves a valores
- Queue – fila

List

- É uma Collection ordenada que pode conter elementos duplicados;
- Pode ser ordenado;
- É implementada por várias classes incluindo: ArrayList, LinkedList e Vector;
- ArrayList e Iterator;
- Vector;

Set

- É uma Collection não ordenada e que não pode conter elementos duplicados;
- Não Pode ser ordenado;
- É implementada por várias classes incluindo: HashSet, JobStateReasons, LinkedHashSet, TreeSet ;
- Iterator;

Map

- É uma Collection que associa chaves a valores;
- Não pode ter mais de um elemento com a mesma chave;
- É implementada por várias classes incluindo:HashMap, TreeMap;
- Iterator;

Properties

Algoritimos de coleções

- Interface Comparable;
- Metodo compareTo;
- Interface Comparator<Objeto>;
- Método compare
- Algoritimos sort, shuffle
- Exemplos;

Stack

- Pilha – ultimo que entra é o primeiro que sai;
- Classe Stack
- Método isEmpty();
- Método push();
- Método pop();
- Método size();

Queue – JSE 5.0

- Fila – primeiro que entra, primeiro que sai;
- Classe PriorityQueue implementa a interface Queue;
- Método offer();
- Método size();
- Método peek();
- Método poll();

Duvidas?

AMBIENT
INFORMÁTICA

Mais do pacote java.util

- Date
- Calendar

Date

- Representa um instante no tempo com precisão de milisegundos;
- Alguns métodos já estão obsoletos;

Calendar

- Calendar é uma classe abstrata;
- Método getInstance();
- Método get();
- Campos:
- HOUR, MINUTE, DAY, etc...

O pacote java.text

- SimpleDateFormat

A classe String

- Não pode ser modificado
- Construtores;
- Comparando Strings;
- Métodos:
  - length(), charAt(), getChars(), indexOf(), lastIndexOf, substring(), toUpperCase, toLowerCase, split.
- a classe StringTokenizer

A classe StringBuffer

- Pode ser modificado;
- Construtores;
- Método append()
- Método capacity()
- Método insert()
- Método reverse()

Saída Formatada

- Formatando a saída com printf
- Arredonda valores de ponto flutuante;
- Alinha uma coluna de números;
- Exibe datas e horas em vários formatos;
- Exibe a quantidade de casas decimais desejadas;

Caracteres de conversão:

- %
- -d; o; x – exibem inteiros;
- -f – exibe ponto flutuante;
- t – exibem datas e horas em vários formatos; utiliza sufixos de conversão:
- H, M, S – hora, minuto, segundo;
- F, Y, m, d – data formatada, ano, mês e dia
- Outros...

Largura e precisão de campos

- %4d – imprime um inteiro em um campo de tamanho 4 alinhado à direita;
- %-4d – imprime um inteiro em um campo de tamanho 4 alinhado à esquerda;
- %.2f – imprime um ponto flutuante com aproximação de duas casas;

Largura e precisão de campos

- Largura e precisão de campos podem ser combinados:
- %4.3f - imprime um ponto flutuante com precisão de três casas em um campo com tamanho igual a 4;

Largura e precisão de campos...

- %,.2f – imprime um ponto flutuante com aproximação de duas casas e separação de milhares;
- Ex.: printf(“%,.2f”, 123456789.11111) imprime: 123.456.789,11

Imprimindo com Índice

- %1$2d, %1$s – indica que todos os especificadores de formato utilizam o primeiro argumento na lista de argumentos. Utilizando índice a ordem dos argumentos não precisa vir necessariamente na ordem dos especificadores;

Outros caracteres de conversão

- %% – imprime o caractere %
- %n – imprime um separador de linha

Método format da classe String

- Retorna uma String formatada com os parâmetros de formatação vistos anteriormente;
- Exemplos:

Dúvidas?

AMBIENT
INFORMÁTICA

Fim

AMBIENT
INFORMÁTICA