**LAPORAN PRAKTIKUM**

**PERTEMUAN-11**

Diajukan untuk memenuhi salat satu tugas praktikum Mata kuliah Pemrograman Berorientasi Objek



**Disusun Oleh:**

**Daiva Raditya Pradipa (231511039)**

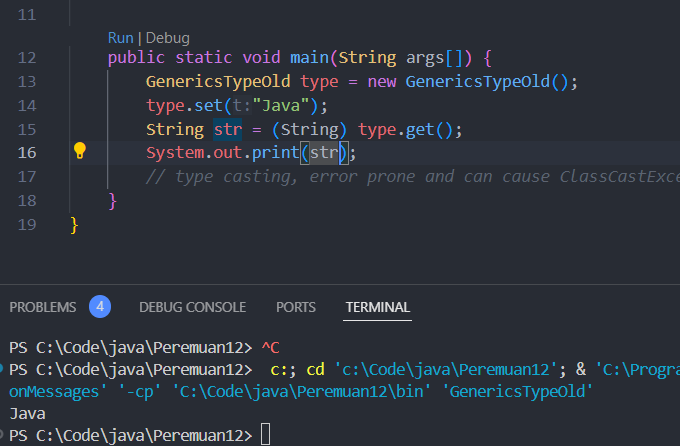
**Jurusan Teknik Komputer dan Informatika**

**Program Studi D-3 Teknik Informatika**

**Politeknik Negeri Bandung**

**2024**

1. **GenericTypeOld**
2. Screenshot hasil program

****

1. Penjelasan

Pada program ini class GenericTypeOld yang memiliki atribut t dengan tipe object, kemudian terdapat method getter dan setter untuk mengambil dan manipulasi atribut t tersetbu. Kemudian terdapat main program untuk meninstansikan class GenericTypeOld menjadi sebuah object. Kemudian terdapat pemanggilan method set pada object type yang dengan parameter string “java”, mengapa hal ini bisa dilakukan meskpiun pada deklarasi method parameter method ini bertipe object ? Hal ini dikarenakan dalam Java, semua kelas adalah subclass dari Object, termasuk String. Karena itu, String bisa digunakan sebagai argumen untuk parameter Object. Kemudian terdapat typecasting untuk mengasign nilai dari t pada class GenericTypeOld ke variabel dengan tipe string menggunakan type casting string.

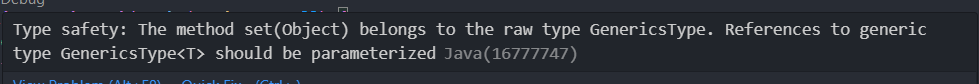
1. Masalah
2. Solusi
3. Teman Yang Membantu
4. **GenericsType**
5. Screenshot hasil program



1. Penjelasan

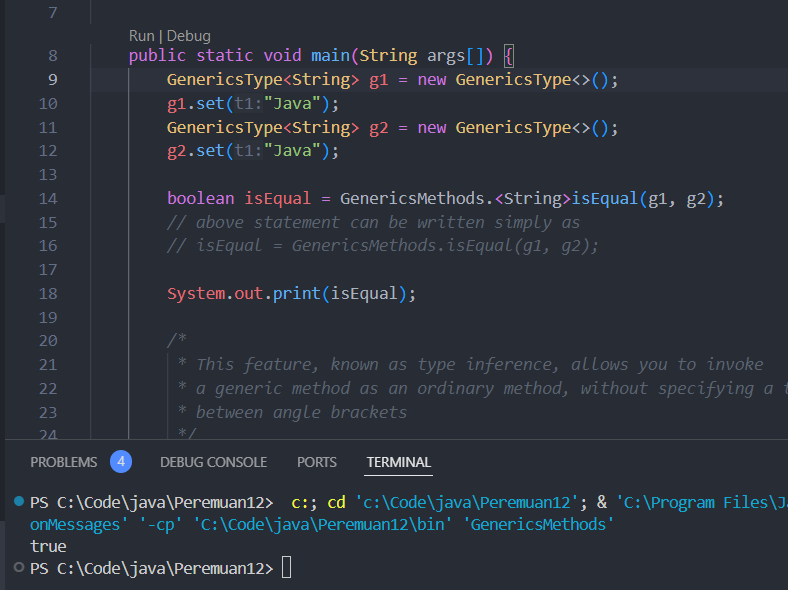
Pada kelas GenericsType, implementasinya mirip dengan kelas GenericsTypeOld sebelumnya. Perbedaannya adalah pada kelas ini diterapkan *generic programming* menggunakan *generic class*. Penerapan *generic programming* ditandai dengan adanya sintaks <T> setelah deklarasi nama kelas. Hal serupa juga terjadi pada atribut t yand dideklarasikan bertipe generil yang secara otomatis membuat konstruktor dan method getter dan setter untuk atribut t di dalam kelas tersebut menjadi bertipe generik.

Kemudian pada main program terdapat instansiasi class GenericsType menjadi sebuah class namun ditambahkan dengan tipe generics String yang yang membuat class object hasil instansiasi memiliki atribut t bertipe string, method set dengan parameter bertipe string, dan method get yang akan mengembalikan nilai string. Maka dari itu saat object memanggil method set dengan mengisikan nilai parameter dengan nilai string tidak terjadi error. Kemudian terdapat instansiasi class GenericsType Kembali menjadi object, namun kali ini tanpa mendefeinisikan tipe generiknya. Hal ini membuat atribut t besert method set dan get dapat memiliki tipe apapun. Contoh pada line code dibawahnya terdapat pemanggilan method set pertama dengan mengisikan parameter dengan nilai string dan pemanggilan method set kedua dengan mengisikan parameter dengan nilai integer atau number. Hasilnya program dapat berjalan dan menghasilkan output dengan nilai 10 saat method get dipanggil. Namun, terdapat peringatan berupa



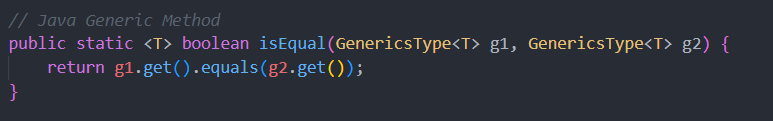
Yang mengindikasikan bahwa method tersebut menggunakan kelas generik tanpa menentukan tipe parametriknya, yang dapat mengurangi type safety atau keamanan tipe. Hal ini dikarenakan kompilator tidak bisa memverifikasi tipe data yang di-set atau diambil dari objek tersebut.

1. Masalah
2. Solusi
3. Teman Yang Membantu
4. **GenericsMethods**
   1. Screenshot hasil program

****

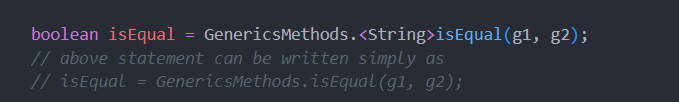
* 1. Penjelasan

Pada class GenericsMethod terdapat method isEqual yang dideklarasikan sebagai berikut



Di sini, kita dapat melihat bahwa metode isEqual adalah metode generik, yang ditandai dengan sintaks <T> sebelum deklarasi method. Terdapat pula arameter g1 dan g2 bertipe GenericsType<T>, di mana T adalah tipe generik yang akan ditentukan saat metode ini dipanggil.

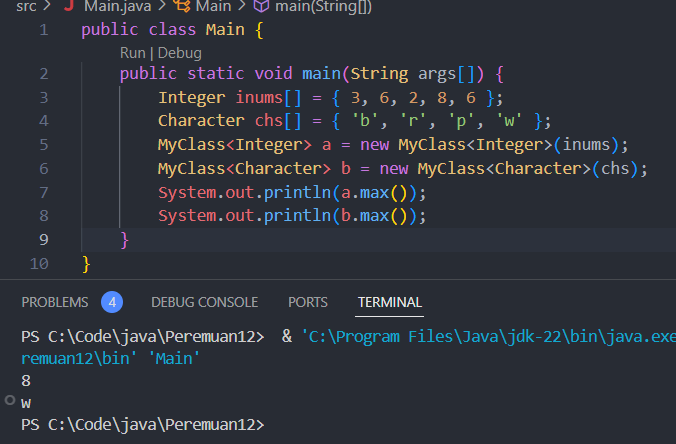
Kemudian terdapat main program terdapat instansiasi dua object dari class GenericsType dengan tipe generics string menjadi object g1 dan g2. Kemudian g1 dan g2 memanggil method set dan masing masing diisikan nilai parameternya dengan nilai string “java”. Kemudian terdapat pemanggilan method isEqual dari class GenericsMethod ini sebagai berikut



Disini kita bisa lihat bahwa pemanggilan method isEqual ini diikuti dengan pendefinisian tipe generic pada method tersebut yaitu string. Meskipun begitu, pemanggilan method ini dapat disimplifikasi menjadi GenericsMethods.isEqual(g1,g2) dengan hasil yang sama untuk masing masing cara pemanggilan yaitu true.

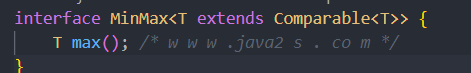
* 1. Masalah
  2. Solusi
  3. Teman yang membantu

1. Java Generic Interface
   1. Screenshot hasil program



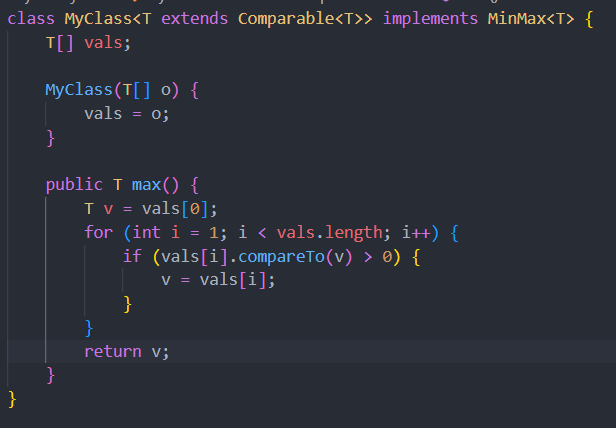
* 1. Penjelasan

Pada program Java Generic Interface ini terdapat sebuah interface dengan nama MinMx sebagai berikut



Disini kita bisa lihat bahwa interface ini merupakan interface generic ditandai dengan adanya syntax <T setelah nama interface yang mengekstend ke class Comparable yang merupakan sebeuah generic class pula. Lalu pada interface ini terdapat method max dengan tipe T atau dapat kita sebut method generic.

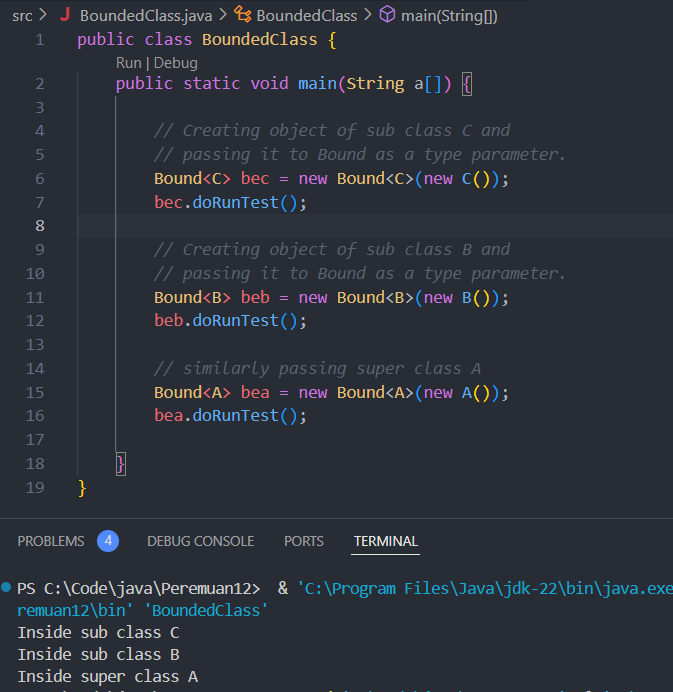
Kemudian terdapat pula class MyClass yang mengimplementasikan interface MinMax ini



Class MyClass disini juga merupakan class generic ditandai dengan syntax<T> seleah deklarasi nama class yang mengekstend ke class generic Comparable dan mengimplement generic interface MinMax. Pada class ini terdapat atribut generic array dengan nama vals beserta constructor dengan tipe parameter array generic. Kemudian terdapat method generic max() yang berfungsi mencari nilai terbesar dalam array vals dengan menggunakan method compateTo dari class comparable.

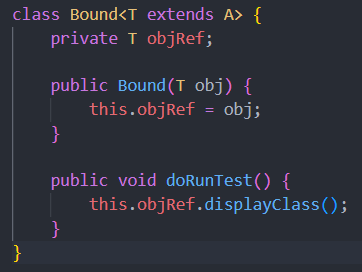
Kemudian pada main program terdapat instansiasi class MyClass ini menjadi 2 objek dengan tipe generic character dengan parameter variabel chs dengan Kumpulan nilai elemen karakter dan integer dengan parameter variable inums yang merupakan array dengan Kumpulan elemen integer. Yang kemudian masing masing objek memanggil method max yang menghasilkan output 8 dan w. Method max ini dapat dipanggil masing masing objek meskipun tanpa mendefinisikan tipe generiknya karena dia otomatis mementukan tipe methodnya berdasarkan input pada saat intansiasi class menjadi objek.

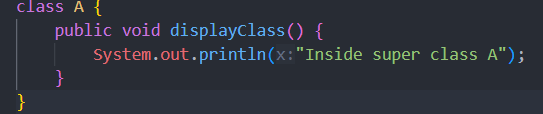
1. Java Bounded Class
   1. Screenshot hasil program



* 1. Penjelasan

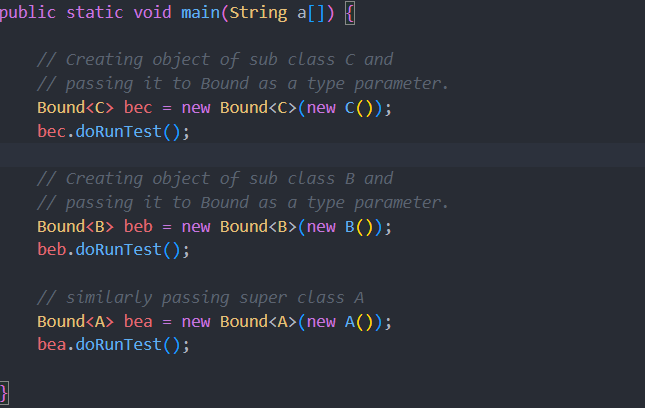
Pada program ini terdapat class Bound yang merupakan class generik yang mengekstend ke class A





Class Bound memeiliki atribut objRef yang merupakan atribut dengan tipe generic yang membuat otomatis constructor juga memiliki parameter dengan tipe generic. Kemudian terdapat method void doRunTest() yang akan memanggil method displayClass() dari atribut objRef.

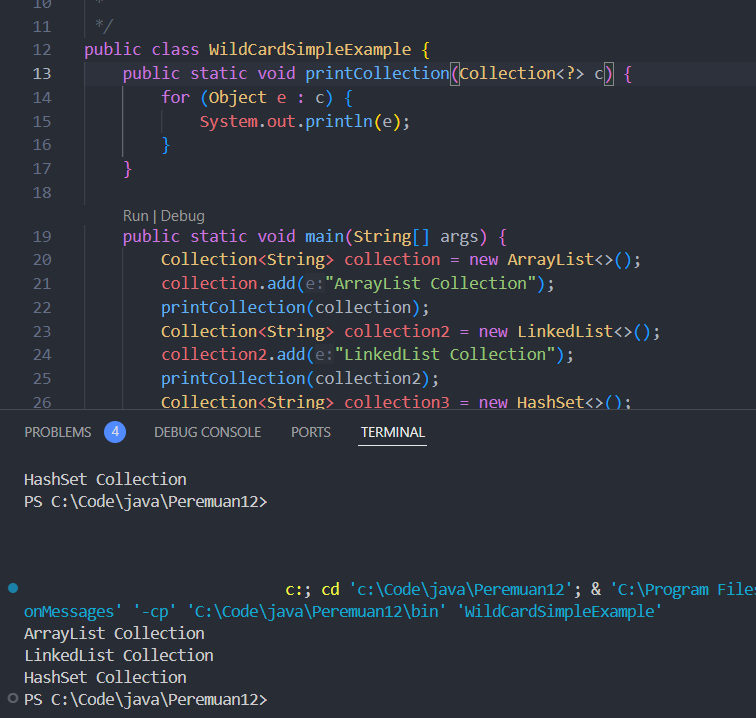
Kemudian pada main program terdapat instansiasi 3 objek dari class Bound ini dengan implementasi sebagai berikut



Dimana masing masing objek merupakan instansiasi dengan tipe mendefinisikan tipe generic yaitu berupa class A, B, dan C dan parameter berupa object dari class C, B, dan A. Sehingga nilai yang bisa di assing atau dikirimkan pada parameter dibatasi tipenya yaitu berupa objek class dari class A, B, dan C. Kemudian masing masing objek menjalankan method doRunTest(0 yang dimana implementasi method ini adalah memanggil method displayClass yang ada pada masing masing class A, B, dan C melalui objek yang didismpan pada atribut class Bound.

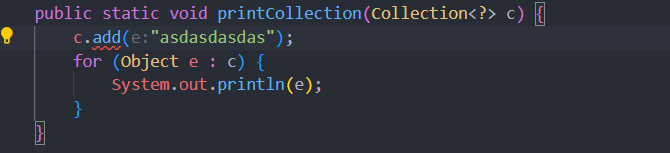
* 1. Masalah
  2. Solusi
  3. Teman yang membantu

1. Java Generic WildCard
   1. Screenshot hasil program

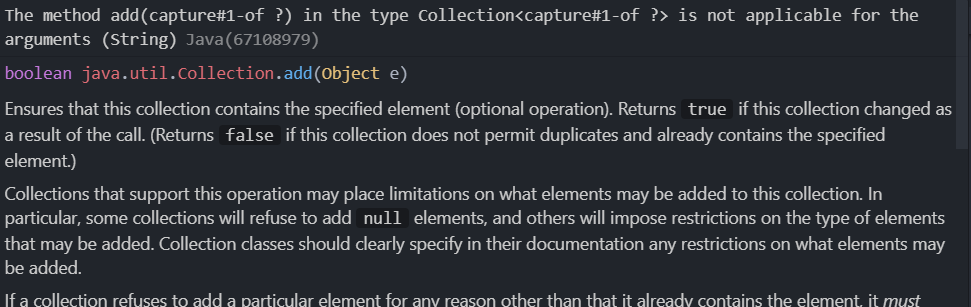


* 1. Penjelasan

Pada class WildCardSimpleExample terdapat method static pringColection yang memeiliki parameter Collection dengan tipe *unbounded wildcard* ditandai dengan syntax <?>. Deklarasi parameter generic dengan sintaks <?> disebut sebagai *unbounded wildcard* dalam bahasa pemrograman Java. Ini berarti bahwa tipe yang diizinkan untuk parameter tersebut bisa merupakan tipe apapun, tanpa batasan spesifik. Selain itu apabila kita mendefinisikan *unbounded wildcard* pada suatu method sebagai contoh pada method printCollection() kita tidak dapat menambahkan element baru koleksi yang dikirimkan melalui parameter sebagai berikut



Disini kita bisa lihat bahwa apabila kita mencoba menambah elemen koleksi di dalam method printCollection maka akan muncul peringatan error yang menyatakan bahwa



Kemudian pada main method, terdapat tiga jenis koleksi berbeda didefinisikan: ArrayList, LinkedList, dan HashSet, masing-masing bertipe String. Setiap koleksi ditambahkan satu elemen teks yang mencantumkan jenis koleksinya. Kemudian, metode printCollection dipanggil untuk masing-masing koleksi, yang mencetak elemen-elemen dari ketiga koleksi ini secara terpisah.

* 1. Masalah
  2. Solusi
  3. Nama teman yang membantu