



Pengajar:

Denny, Satrio Baskoro Yudhoatmojo,

Muhammad Yusuf Khadafi, Bayu Distiawan

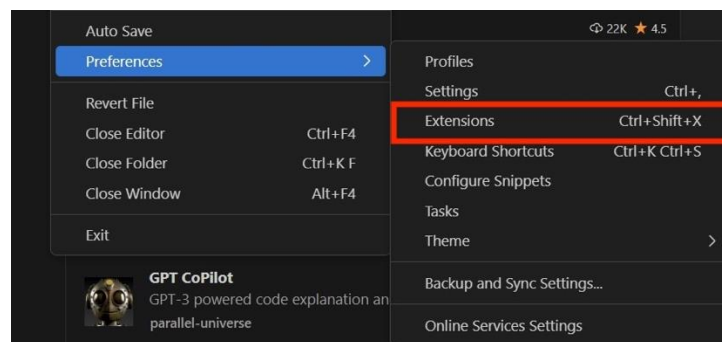
Trisedya

Peraturan dan Informasi:

1. Pastikan Anda sudah mengikuti semua instruksi pada dokumen Persiapan UTS APAP.
2. Pastikan Anda sudah mengisi Pernyataan Patuh terhadap Tata Tertib dan Kode Etik Ujian di <https://scele.cs.ui.ac.id/mod/questionnaire/view.php?id=180471>. UTS Anda tidak dinilai jika Anda tidak mengisi pernyataan ini.
3. Sifat ujian adalah **Open All**, tetapi segala bentuk komunikasi **tidak** diperbolehkan saat ujian. Silakan gunakan semua resources atau contoh-contoh kode saat mengerjakan tutorial/tugas yang pernah Anda kembangkan saat UTS. Jadi, Anda tidak perlu mengembangkan kode Anda dari nol.
4. Anda diperbolehkan menggunakan mesin pencari seperti *Google* atau berselancar di Internet jika mengalami kesulitan saat membuat program. Namun, Anda tidak diperbolehkan menggunakan ChatGPT. **Sanksi jika menggunakan ChatGPT/AI adalah nilai akhir E. Jika fitur Copilot terinstall di editor Anda, disable/uninstall fitur tersebut dan tunjukkan ke pengawas ujian di awal waktu UTS mulai. Jika Anda diminta menunjukkan halaman konfigurasi/preferensi dan ternyata fitur terkait AI aktif, maka Anda dianggap melakukan kecurangan.** Segala kecurangan akan dilaporkan dan ditindak tegas.

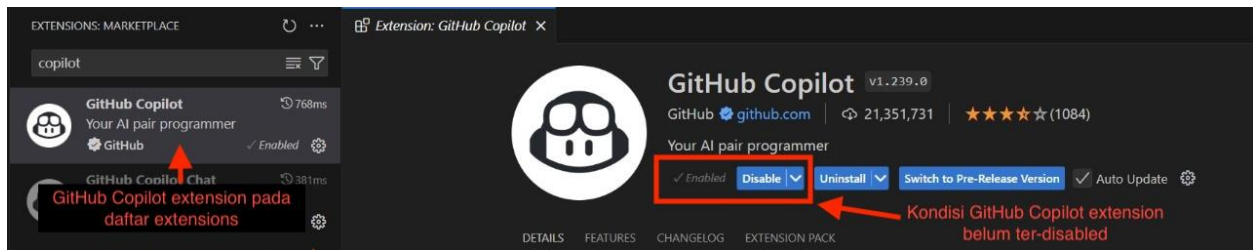
4.1. Instruksi *Disabling* Fitur GitHub Copilot pada Visual Studio Code.

- Pilih menu **File->Preferences->Extensions** untuk mencari GitHub Copilot extension yang terpasang di Visual Studio Code Anda (lihat Gambar 1).

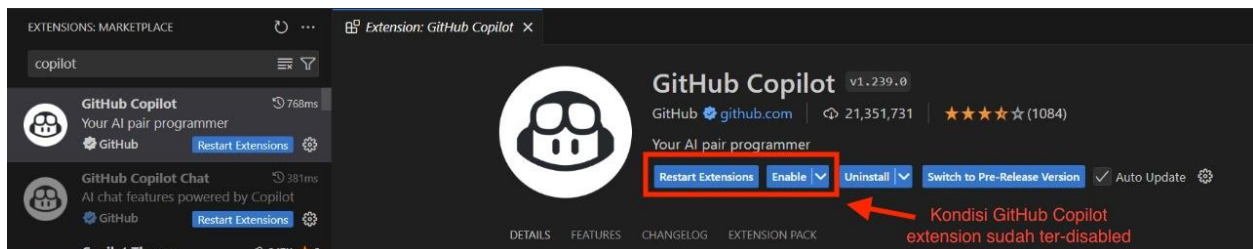


Gambar 1 Menu Extensions pada Visual Studio Code.

- Ketuk **Github Copilot** pada daftar extension di Visual Studio Code Anda (lihat Gambar 2) dan pilih **Disable** pada halaman deskripsi Github Copilot extension. Jika sudah ter-*disabled* maka tampilan panel deskripsi GitHub Copilot akan seperti Gambar 3.



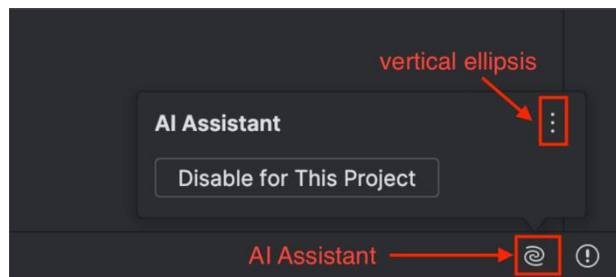
Gambar 2 Tampilan GitHub Copilot extension pada daftar extensions Visual Studio Code dan panel deskripsi GitHub Copilot dengan informasi status extension masih enabled.



Gambar 3 Tampilan GitHub Copilot extension dengan kondisi extension tersebut sudah di-disabled.

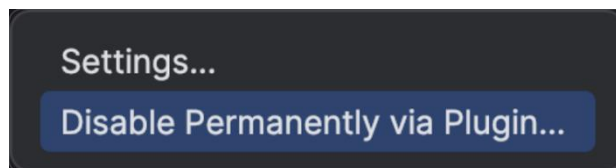
4.2. Contoh Instruksi *Disabling* Fitur AI Assistant pada IntelliJ IDEA.

- Ketuk fungsi **AI Assistant** yang terletak di pojok kanan bawah (lihat Gambar 4).



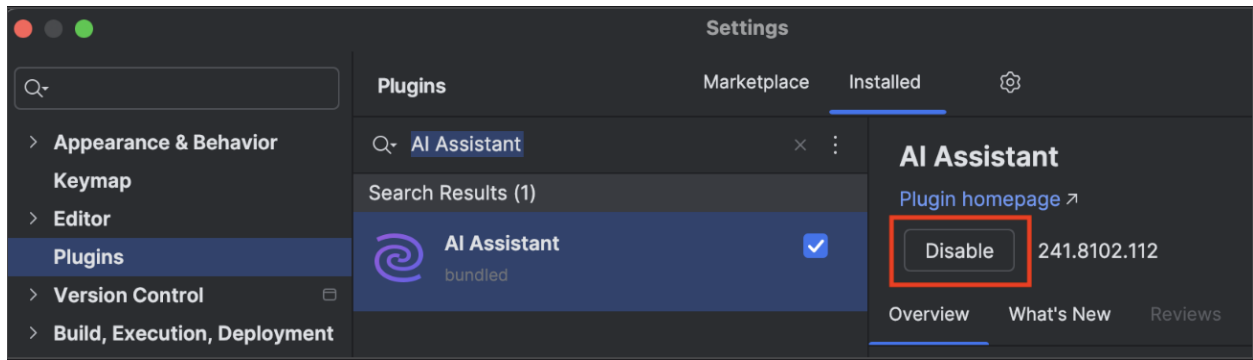
Gambar 4 Letak fungsi AI Assistant pada IntelliJ IDEA serta ikon vertical ellipsis.

- Ketuk ikon **vertical ellipsis (More)** (lihat Gambar 4) dan pilih pilihan **Disable Permanently via Plugin** (Gambar 5).



Gambar 5 Tampilan pilihan **Disable Permanently via Plugin** yang muncul setelah ikon vertical ellipsis diketuk.

- Pada panel deskripsi dari *plugin* tersebut yang terbuka setelah dipilih, ketuk **Disable** (lihat Gambar 6) dan kemudian *restart* IntelliJ IDEA. Jika ternyata panel AI Assistant *plugin* belum terbuka, maka Anda dapat mencari nama *plugin* di atas pada **Settings | Plugins | Installed**.

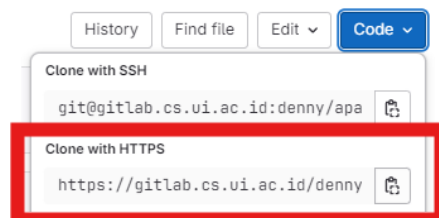


Gambar 6 Tampilan panel deskripsi dari AI Assistant plugin. Pastikan AI Assistant sudah disabled seperti yang terlihat pada kotak merah di gambar.

5. Alat komunikasi dan *gadget* lain dimatikan dan dimasukkan ke dalam tas. Tas Anda harus diletakkan di depan ruangan kelas. Di atas meja, hanya ada alat tulis dan laptop.
6. Jika Anda tidak mempunyai laptop yang memadai, Anda dapat menggunakan PC di laboratorium. Anda diperbolehkan datang lebih awal untuk mempersiapkan PC Anda.
7. **Anda boleh memberikan lebih dari yang diminta di soal 😊**
8. **Total nilai adalah 105, di mana 5 adalah nilai bonus.**
9. **Jika ada penjelasan soal yang kurang, Anda boleh berasumsi atau berinovasi.**

Submission:

1. Pada saat ujian, Anda diminta untuk push semua commit Anda ke repository Anda sendiri (git push origin main) yang sudah dibuat pada saat persiapan UTS.
 - a. **Satu commit hanya berisi perubahan terkait satu soal saja. Sebuah commit yang berisi perubahan terkait beberapa soal, tidak dinilai.**
 - b. Setiap soal **boleh** terdiri dari beberapa commit. Namun **lebih baik** jika hanya 1 commit per soal.
 - c. Commit message harus mengikuti pola: "solve #99: catatan ringkas. Catatan rinci." dimana **99** adalah **nomor soal/fitur**.
 - d. Kerjakan Soal #0 dahulu. Kemudian, urutan commit tidak harusurut sesuai nomor urut soal.
 - e. Hanya **branch "main"** yang akan dinilai.
 - f. Anda diminta untuk **push ke repository Anda (origin) secara berkala**.
2. Pastikan Anda sudah melaporkan URL repository Anda (**yang HTTPS, bukan SSH**) di SCELE pada Questionnaire "UTS APAP 2024 survey" <https://scele.cs.ui.ac.id/mod/questionnaire/view.php?id=180441>. Anda tidak perlu submit lagi jika sudah melakukannya.



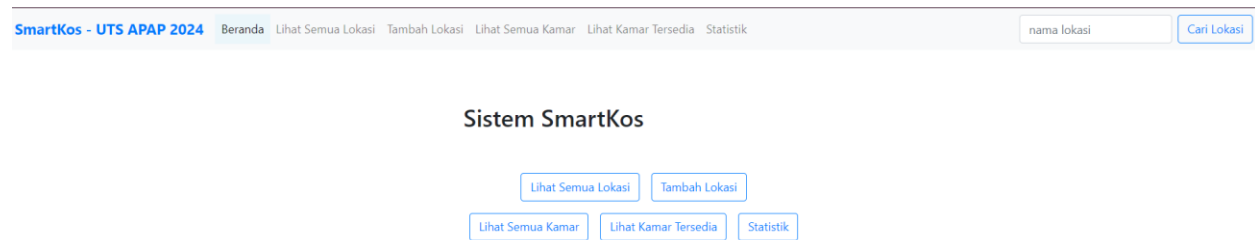
3. Anda harus mengunggah *source code* ke SCeLE. **Jangan mengunggah folder build!**

Sebelum mengunggah ke SCeLE, lakukan:

- a. “**./gradlew.bat clean**” (Windows) atau “**bash gradlew clean**” (Linux/Mac) untuk menghapus folder **build**. Kemudian hapus/tidak sertakan directory '**db**'.
- b. Kompres folder *source code* dengan **zip**. Format penamaan file UTS24_[KELAS]_[NAMA DEPAN]_[NPM].zip. Contoh UTS24_A_BUDI_1234567.zip
- c. Setelah itu, unggah/upload ke dropbox yang disediakan di SCeLE <https://scele.cs.ui.ac.id/mod/assign/view.php?id=180445>

Smart Kos – Sistem Manajemen Kamar Kost

APAP Kost merupakan sebuah perusahaan penyewaan kamar kost memiliki kendala dalam mengorganisir banyak kamar kost yang tersebar di banyak lokasi, Anda sebagai seorang pengembang software diminta untuk membuat sebuah sistem manajemen kamar kost yang diharapkan mampu mempermudah APAP Kost dalam mengelola data kamar, data lokasi serta informasi statistik sehingga membuat operasional dan pemeliharaan kamar yang dimiliki dapat menjadi lebih efisien.



Penting!

Sebelum Anda coding, Anda dapat pull template UTS yang lebih lengkap, seperti halaman muka di atas. Pastikan Anda telah menambahkan remote repository "upstream" di repository git lokal Anda sesuai dengan panduan persiapan UTS.

Buka terminal/command prompt pada directory repository git lokal Anda.

Jalankan:

```
> git pull upstream main
> git push origin main
```

Kemudian, build dan run source code ini untuk memastikan konfigurasi sudah berjalan dengan baik.

Pelajari source code yang sudah tersedia, seperti Model, Controller, Service, template, fragment

Berikut adalah fungsionalitas yang harus Anda implementasikan dalam tugas ini. Baca detail dari tugas ini pada halaman berikutnya.

Feature Nomor	Tugas	Nilai
0.	Update NPM, first commit, push ke gitlab	5
1.	Menampilkan daftar semua lokasi dalam bentuk tabel menggunakan Thymeleaf. (View All)	15
2.	Menambah lokasi baru	10
3.	Menambahkan kamar baru pada sebuah lokasi	10
4.	Update status kamar antara 'tersedia', 'disewakan', dan 'tidak tersedia'	10
5.	Implementasi cari Lokasi menggunakan nama lokasi	10
6.	Menampilkan daftar kamar yang tersedia saja	10
7.	Statistik kamar per lokasi dan kamar tersedia menggunakan DataTables (10 point) Anda juga harus mengimplementasikan REST API yang dibutuhkan (10 point)	20
8.	Kolom Estimasi Uang Sewa Pertahunan.	10
9.	Terkait dengan Dependency injection, Anda diminta mengubah dari Setter dependency injection/field injection ke Constructor Dependency Injection	5

Fitur #0. Update NPM, First Commit, Push ke Gitlab

Ganti angka **1234567890** pada file file: src/main/resources/templates/home.html dengan NPM Anda. Kemudian, buatlah sebuah commit dengan message "solve #0: fix NPM". Lalu push commit Anda ke gitlab repository Anda.

```
<!DOCTYPE html>
<html lang="en" xmlns:th="http://thymeleaf.org">
<head>
  <meta charset="UTF-8">
  <title>UTS - 1234567890</title>
  <object th:insert=~{fragments/fragment::css}" th:remove="tag"></object>
  <object th:insert=~{fragments/fragment::js}" th:remove="tag"></object></head>
<body>
```

Contoh perintah pada command line:

```
> git commit -m "solve #0: fix NPM"
> git push origin
```

Fitur #1. Melihat Daftar Ruangan

URL mapping: `http://<domain_name>:8080/lokasi/view-all`

SmartKos - UTS APAP 2024 Beranda **Lihat Semua Lokasi** Tambah Lokasi Lihat Semua Kamar Lihat Kamar Tersedia Statistik

nama lokasi

Id	Nama Lokasi	Alamat	Kota	Action
1	Kos Mawar	Jl. Mawar No. 10	Depok	<input type="button" value="Detail"/>
2	Kos Melati	Jl. Melati No. 5	Bandung	<input type="button" value="Detail"/>
3	Kos Anggrek	Jl. Anggrek No. 2	Surabaya	<input type="button" value="Detail"/>
4	Kos Kenanga	Jl. Kenanga No. 8	Yogyakarta	<input type="button" value="Detail"/>
5	Kos Flamboyan	Jl. Flamboyan No. 12	Semarang	<input type="button" value="Detail"/>

Berikut adalah informasi mengenai data lokasi yang ingin dicatat.

```
@Table(name = "lokasi")
public class LokasiModel implements Serializable {
    @Id
    @GeneratedValue
    @SequenceGenerator
    @Column(name = "lokasi_id")
    private long lokasiId;
    @Column(name = "nama_lokasi")
    private String namaLokasi;
    @Column(name = "alamat")
    private String alamat;
    @Column(name = "kota")
    private String kota;
}
```

Untuk **fitur #1**, Anda diminta untuk membuat tampilan seperti yang ada pada gambar di atas, yaitu menampilkan semua daftar lokasi yang ada di basis data. Ketentuan dari fitur ini adalah:

- Jika tombol “Detail” diklik, tampilan akan berpindah ke **View Detail** untuk melihat detail lokasi tersebut (lihat **fitur #3**).

Keterangan:

User interface tidak perlu sama persis dengan contoh di atas. Anda bebas berkreasi. Namun pastikan Anda menampilkan data yang diminta di soal.

Fitur #2. Menambah data Lokasi baru

URL mapping: **http://localhost:8080/lokasi/add**

SmartKos - UTS APAP 2024 Beranda Lihat Semua Lokasi **Tambah Lokasi** Lihat Semua Kamar Lihat Kamar Tersedia Statistik

nama lokasi Cari Lokasi

Tambah Lokasi

Nama lokasi

Alamat

Kota

Untuk **fitur #2**, Anda diminta untuk membuat tampilan seperti gambar di atas, yaitu membuat tampilan untuk menambah lokasi baru.

Note : untuk dropdown kota berisi nilai : "Depok", "Jakarta Pusat", "Bandung", "Semarang", "Yogyakarta"

Jika Button **Submit** di-klik, tampilan akan kembali ke **View All**. Berikut adalah contoh perubahan halaman. Perhatikan bahwa terdapat pesan bahwa lokasi baru berhasil di simpan pada halaman ini. **Hint:** RedirectAttributes, Flash Message.

SmartKos - UTS APAP 2024 Beranda **Lihat Semua Lokasi** Tambah Lokasi Lihat Semua Kamar Lihat Kamar Tersedia Statistik

nama lokasi Cari Lokasi

Lokasi berhasil disimpan dengan id 1000

Id	Nama Lokasi	Alamat	Kota	Action
1	Kos Mawar	Jl. Mawar No. 10	Depok	<input type="button" value="Detail"/>
2	Kos Melati	Jl. Melati No. 5	Bandung	<input type="button" value="Detail"/>
3	Kos Anggrek	Jl. Anggrek No. 2	Surabaya	<input type="button" value="Detail"/>
4	Kos Kenanga	Jl. Kenanga No. 8	Yogyakarta	<input type="button" value="Detail"/>
5	Kos Flamboyan	Jl. Flamboyan No. 12	Semarang	<input type="button" value="Detail"/>
1000	Cikini City	Jl Cikini Raya	Jakarta Pusat	<input type="button" value="Detail"/>

Keterangan:

User interface tidak perlu sama persis dengan contoh di atas. Anda bebas berkreasi. Namun pastikan Anda menampilkan data yang diminta di soal.

Fitur #3. Menambahkan kamar baru pada sebuah lokasi

URL mapping: <http://<domain name>:8080/lokasi/{lokasiId}/view>

SmartKos - UTS APAP 2024

Beranda

Lihat Semua Lokasi

Tambah Lokasi

Lihat Semua Kamar

Lihat Kamar Tersedia

Statistik

nama lokasi

Cari Lokasi

Detail Lokasi

Id Lokasi

2

Nomor/Kode Lokasi

Kos Melati

Nama Lokasi

Jl. Melati No. 5

Kota

Bandung

Submit

Back

Daftar Kamar

Id Kamar	Nomor Kamar	Lantai	Harga Sewa	Status	Action
6	101	1	2,000,000	tersedia	<div>Update Status</div>
7	102	1	2,200,000	disewakan	<div>Update Status</div>
8	201	2	2,100,000	tersedia	<div>Update Status</div>

Tambah Kamar

Nomor kamar

nomor kamar

Lantai

lantai

Harga sewa

0

Status

tersedia

Submit

Back

Fitur ini diakses melalui tombol detail dari halaman view-all lokasi

Dalam fitur ini, Anda diminta untuk dapat menambahkan kamar untuk lokasi yang di pilih. Untuk setiap lokasi bisa memiliki banyak kamar.

Status kamar pada dropdown status adalah "tersedia", "disewakan", dan "tidak tersedia"

Anda harus menggunakan class **KamarModel** yang sudah disediakan. Anda diperbolehkan untuk mengubah class **KamarModel** yang diberikan.

Perhatikan juga format angka dengan *thousand separator* dan rata kanan.

Fitur #4. Update Status Kamar

URL mapping: <http://<domain name>:8080/kamar/{kamarId}/update>

SmartKos - UTS APAP 2024 Beranda Lihat Semua Lokasi Tambah Lokasi Lihat Semua Kamar Lihat Kamar Tersedia Statistik nama lokasi Cari Lokasi

Update Status Kamar

nomorKamar

lantai

hargaSewa

Status

Fitur ini dapat diakses dari tombol update status pada daftar kamar

Untuk **fitur #4**, Anda diminta untuk membuat tampilan seperti gambar di atas, yaitu membuat tampilan untuk update status kamar. URL mapping mengandung **Path Variable kamarId**.

Ketika button **Submit** di-klik, status dengan id tersebut akan diubah sesuai dengan informasi baru yang dimasukkan lewat form. Tampilan kemudian akan berpindah ke halaman **Lihat Semua Kamar**. Perhatikan bahwa terdapat pesan bahwa status berhasil diubah pada halaman View All seperti pada fitur sebelumnya.

SmartKos - UTS APAP 2024 Beranda Lihat Semua Lokasi Tambah Lokasi **Lihat Semua Kamar** Lihat Kamar Tersedia Statistik nama lokasi Cari Lokasi

Status berhasil di update

Lokasi	Id Kamar	Nomor Kamar	Lantai	Harga Sewa	Status	Action
Kos Mawar	1	101	1	1,500,000	tersedia	<input type="button" value="Update Status"/>
Kos Mawar	2	102	1	1,800,000	disewakan	<input type="button" value="Update Status"/>
Kos Mawar	3	201	2	1,600,000	tersedia	<input type="button" value="Update Status"/>
Kos Mawar	4	202	2	1,700,000	disewakan	<input type="button" value="Update Status"/>
Kos Mawar	5	301	3	1,500,000	tersedia	<input type="button" value="Update Status"/>
Kos Melati	7	102	1	2,200,000	disewakan	<input type="button" value="Update Status"/>
Kos Melati	8	201	2	2,100,000	tersedia	<input type="button" value="Update Status"/>
Kos Kenanga	9	101	1	1,800,000	tersedia	<input type="button" value="Update Status"/>
Kos Kenanga	10	202	2	1,900,000	disewakan	<input type="button" value="Update Status"/>

Keterangan:

User interface tidak perlu sama persis dengan contoh di atas. Anda bebas berkreasi. Namun pastikan Anda menampilkan data yang diminta di soal.

Fitur #5. Cari Lokasi menggunakan Nama Lokasi

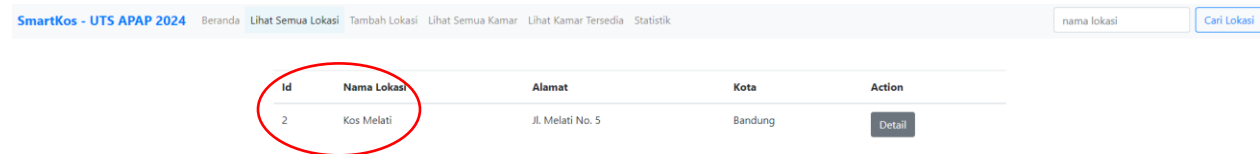
Pengguna dapat mengisi nama lokasi yang dicari pada menu bar di sebelah kanan atas. Pencarian ini harus bersifat **case insensitive** dan juga dapat berupa **substring**.



Sistem SmartKos



Contoh: Setelah mengetik “melati” pada text box dan klik tombol “Cari Lokasi”, maka ditampilkan semua lokasi dimana nama lokasi-nya mengandung string “melati”.



Fitur #6. Daftar Kamar Tersedia

URL mapping: <http://<domain name>:8080/kamar/tersedia>

Fitur ini menampilkan informasi kamar dengan status tersedia.

Perhatikan juga format angka dengan *thousand separator* dan rata kanan.

SmartKos - UTS APAP 2024

Beranda

Lihat Semua Lokasi

Tambah Lokasi

Lihat Semua Kamar

Lihat Kamar Tersedia

Statistik

nama lokasi

Cari Lokasi

Lokasi	Id Kamar	Nomor Kamar	Lantai	Harga Sewa	Status	Action
Kos Mawar	1	101	1	1,500,000	tersedia	<button>Update Status</button>
Kos Mawar	3	201	2	1,600,000	tersedia	<button>Update Status</button>
Kos Mawar	5	301	3	1,500,000	tersedia	<button>Update Status</button>
Kos Melati	8	201	2	2,100,000	tersedia	<button>Update Status</button>
Kos Kenanga	9	101	1	1,800,000	tersedia	<button>Update Status</button>
Kos Kenanga	11	303	3	1,700,000	tersedia	<button>Update Status</button>

Fitur #7. Statistik kamar per lokasi dan kamar tersedia menggunakan DataTables

URL mapping: <http://localhost:8080/report/statistics>

REST API mapping: <http://localhost:8080/api/statistics> dan <http://localhost:8080/api/kamar/tersedia>

SmartKos - UTS APAP 2024

Beranda

Lihat Semua Lokasi

Tambah Lokasi

Lihat Semua Kamar

Lihat Kamar Tersedia

Statistik

nama lokasi

Cari Lokasi

Statistik per Lokasi

10 entries

Search:

Lokasi	Jumlah Kamar Disewakan	Jumlah Uang Sewa Per Bulan	Estimasi Uang Sewa Per Tahunan
Kos Kenanga	1	1,900,000	22,800,000
Kos Mawar	2	3,500,000	42,000,000
Kos Melati	2	4,200,000	50,400,000

Showing 1 to 3 of 3 entries

Previous

1

Next

Kamar Tersedia

10 entries

Search:

Lokasi	Id Kamar	Nomor Kamar	Lantai	Harga Sewa	Status
Kos Kenanga	9	101	1	1,800,000	tersedia
Kos Kenanga	11	303	3	1,700,000	tersedia
Kos Mawar	1	101	1	1,500,000	tersedia
Kos Mawar	3	201	2	1,600,000	tersedia
Kos Mawar	5	301	3	1,500,000	tersedia
Kos Melati	8	201	2	2,100,000	tersedia

Showing 1 to 6 of 6 entries

Previous

1

Next

Library terkait Datatables sudah tersedia di `resources/static/assets/`

Perhatikan juga format angka dengan *thousand separator* dan rata kanan.

Anda juga harus mengimplementasikan REST API yang dibutuhkan. Contoh respon REST API:

API Statistics :

```
▼ [
  ▼ {
    "namaLokasi": "Kos Kenanga",
    "totalUangSewa": 1900000,
    "kamarDisewakan": 1,
    "totalUangSewaTahunan": 22800000
  },
  ▼ {
    "namaLokasi": "Kos Mawar",
    "totalUangSewa": 3500000,
    "kamarDisewakan": 2,
    "totalUangSewaTahunan": 42000000
  },
  ▼ {
    "namaLokasi": "Kos Melati",
    "totalUangSewa": 4200000,
    "kamarDisewakan": 2,
    "totalUangSewaTahunan": 50400000
  }
]
```

API Kamar Tersedia

```
[
  ▼ {
    "kamarId": 1,
    "nomorKamar": "101",
    "lantai": "1",
    "hargaSewa": 1500000,
    "status": "tersedia",
    "lokasiModel": {
      "lokasiId": 1,
      "namaLokasi": "Kos Mawar",
      "alamat": "Jl. Mawar No. 10",
      "kota": "Depok"
    }
  },
  ▼ {
    "kamarId": 3,
    "nomorKamar": "201",
    "lantai": "2",
    "hargaSewa": 1600000,
    "status": "tersedia",
    "lokasiModel": {
      "lokasiId": 1,
      "namaLokasi": "Kos Mawar",
      "alamat": "Jl. Mawar No. 10",
      "kota": "Depok"
    }
  },
  ▼ {
    "kamarId": 5,
    "nomorKamar": "301",
    "lantai": "3",
    "hargaSewa": 1500000,
    "status": "tersedia",
    "lokasiModel": {
      "lokasiId": 1,
      "namaLokasi": "Kos Mawar",
      "alamat": "Jl. Mawar No. 10",
      "kota": "Depok"
    }
  }
],
```

Keterangan:

Respon REST API dan User interface tidak perlu sama persis dengan contoh di atas. Anda bebas berkreasi. Namun pastikan Anda menampilkan data yang diminta di soal.

Fitur #8. Kolom Estimasi Uang Sewa Pertahunan

Anda diminta untuk menambahkan kolom estimasi uang sewa per tahunan pada datatables statistik per lokasi. Perhatikan juga format angka dengan *thousand separator* dan rata kanan.

Anda perlu menjelaskan dalam *commit message* terkait **bagaimana** Anda mengimplementasi ini dan **kenapa**.

SmartKos - UTS APAP 2024 Beranda Lihat Semua Lokasi Tambah Lokasi Lihat Semua Kamar Lihat Kamar Tersedia Statistik [Cari Lokasi](#)

Statistik per Lokasi

Show entries Search:

Lokasi	Jumlah Kamar Disewakan	Jumlah Uang Sewa Per Bulan	Estimasi Uang Sewa Per Tahunan
Kos Kenanga	1	1,900,000	22,800,000
Kos Mawar	2	3,500,000	42,000,000
Kos Melati	2	4,200,000	50,400,000

Showing 1 to 3 of 3 entries [Previous](#) [1](#) [Next](#)

Fitur #9. Dependency Injection

Anda diminta mengubah dari Setter dependency injection/field injection ke Constructor Dependency Injection. Sebaiknya Anda mengerjakan fitur ini setelah fitur-fitur utama diselesaikan.