

Data Analytics for Business 2024

MID EXAM

Anggota Kelompok

- | | | |
|----|-----------------------|---------|
| 1. | Syayid Muhammad Akbar | CS04466 |
| 2. | Farras Daffa Y | CS04172 |
| 3. | Syadira Hanaya | CS04214 |

BAB I

Pendahuluan

1.1 Latar belakang masalah

Industri perhotelan merupakan salah satu sektor penting dalam perekonomian global. Dalam operasionalnya, hotel harus mampu mengelola berbagai aspek yang berkaitan dengan reservasi tamu, tarif kamar, permintaan khusus, serta pembatalan reservasi. Pengelolaan data reservasi dengan baik sangat penting untuk meningkatkan efisiensi dan profitabilitas bisnis. Salah satu tantangan utama yang sering dihadapi oleh manajemen hotel adalah tingginya tingkat pembatalan reservasi, yang berdampak pada potensi kerugian pendapatan dan sumber daya yang terbuang.

Dalam konteks ini, digunakan untuk melakukan analisis terkait berbagai faktor penting yang mempengaruhi operasional hotel. Dataset ini berisi informasi mengenai reservasi dari dua jenis hotel, yakni **City Hotel** dan **Resort Hotel**, dengan rincian tentang lead time (waktu antara reservasi dan kedatangan), status pembatalan, permintaan khusus, dan rata-rata tarif harian (ADR). Informasi ini akan digunakan untuk menyelesaikan beberapa masalah bisnis yang sering dihadapi oleh hotel, khususnya dalam hal pembatalan reservasi dan strategi pengelolaan pendapatan.

1.1.1 Masalah Bisnis yang ingin Diselesaikan

Perusahaan hotel mengalami beberapa masalah dalam proses reservasi yang menyebabkan meningkatnya tingkat pembatalan dan rendahnya permintaan untuk reservasi jangka panjang. Masalah yang telah diidentifikasi antara lain:

1) **Tingkat Pembatalan yang Tinggi:**

Dataset menunjukkan bahwa persentase pembatalan reservasi cukup tinggi (lihat kolom **is_canceled**). Banyak pelanggan membatalkan reservasi mereka setelah proses pemesanan selesai, meskipun kamar telah dialokasikan untuk mereka. Hal ini berdampak negatif terhadap pendapatan hotel dan efisiensi sumber daya.

2) **Lead Time yang Bervariasi:**

Proses reservasi menunjukkan adanya variasi yang signifikan dalam lead time (lihat kolom **lead_time**), yaitu waktu antara pemesanan dan kedatangan pelanggan. Hal ini menyebabkan ketidakpastian dalam pengelolaan ketersediaan kamar, dan dapat

mengakibatkan overbooking atau underbooking yang berdampak pada pengalaman pelanggan.

3) **Permintaan Khusus Tidak Dikelola dengan Baik:**

Permintaan khusus dari pelanggan (lihat kolom **total_of_special_requests**) sering kali tidak dapat dipenuhi dengan baik karena kurangnya mekanisme penanganan yang efektif. Hal ini mengurangi kepuasan pelanggan dan meningkatkan risiko pembatalan reservasi.

4) **Proses Pembayaran yang Tidak Efisien:**

Beberapa pelanggan mengalami kendala dalam proses pembayaran (lihat kolom **adr** dan **reservation_status**). Proses pembayaran yang lambat atau tidak efisien menyebabkan keraguan pelanggan untuk melanjutkan reservasi, sehingga mengurangi potensi pendapatan hotel.

5) **Kurangnya Komunikasi Selama Proses Check-In dan Check-Out:**

Ketidaksempurnaan dalam komunikasi selama proses check-in dan check-out sering terjadi (lihat kolom **reservation_status**). Informasi tidak disampaikan tepat waktu kepada pelanggan, yang menyebabkan keterlambatan dan ketidakpuasan pelanggan terhadap layanan yang diberikan.

1.1.2 Tahapan yang akan dilakukan untuk Penyelesaian Masalah

Berikut adalah tahapan penyelesaian masalah yang akan digunakan:

- 1) Mengidentifikasi tahapan proses reservasi dari awal hingga akhir, menganalisis masalah yang ada berdasarkan data, dan membuat diagram BPMN proses bisnis reservasi yang sedang berjalan.
- 2) Melakukan data preprocessing untuk membersihkan beberapa data yang kotor pada dataset.
- 3) Menganalisis menggunakan SQL Query berdasarkan database database reservasi hotel yang disediakan, untuk mengevaluasi kinerja hotel dan memahami pola reservasi pelanggan.
- 4) Melakukan data exploratory untuk menjelaskan data dalam bentuk grafik.

- 5) Terakhir, melakukan pengujian A/B testing untuk menguji suatu hipotesa yang sudah dibuat.

1.1.3 Output yang akan Dihasilkan

Berikut output yang akan dihasilkan dalam analisis ini:

- 1) Tahapan proses reservasi dari awal hingga akhir, hasil analisis masalah yang ada berdasarkan data, dan gambar diagram BPMN proses bisnis reservasi yang sedang berjalan.
- 2) Menghasilkan data yang sudah bersih
- 3) Menghasilkan hasil analisis evaluasi kinerja hotel dan pola reservasi pelanggan
- 4) Menghasilkan grafik sebagai exploratory data.
- 5) Menghasilkan hipotesa menolak atau menerima H_0 .

BAB II

Business Process Analysis

2.1 Identifikasi Masalah

Berikut adalah identifikasi tahapan proses reservasi dari awal hingga akhir:

1. Memulai Reservasi:

- Pelanggan mengunjungi situs web hotel, mencari kamar yang tersedia, dan mengisi detail tanggal yang diinginkan.

2. Memilih Kamar:

- Pelanggan memilih kamar yang ingin dipesan. Sistem akan memeriksa ketersediaan kamar tersebut.

3. Pembayaran:

- Jika kamar tersedia, pelanggan melanjutkan ke proses pembayaran. Pembayaran ini mencakup deposit, jika diperlukan.
- Pembayaran akan diverifikasi oleh sistem pembayaran digital. Jika pembayaran disetujui, pelanggan akan menerima konfirmasi, dan deposit akan ditahan jika dibutuhkan.
- Jika pembayaran ditolak, reservasi dianggap gagal dan pelanggan diberi tahu.

4. Konfirmasi Reservasi:

- Setelah pembayaran disetujui, sistem mengirimkan konfirmasi kepada pelanggan bahwa reservasi telah diterima dan kamar telah dialokasikan.

5. Permintaan Khusus (Jika Ada):

- Pelanggan dapat memasukkan permintaan khusus untuk kamar atau layanan tertentu. Permintaan ini dicatat dan diproses oleh sistem.
- Hotel akan mempersiapkan permintaan khusus tersebut dan memberikan konfirmasi kepada pelanggan jika permintaan berhasil dipenuhi. Jika tidak, pelanggan akan diberi tahu bahwa permintaan tidak dapat dipenuhi.

6. Check-in di Hotel:

- Pada hari kedatangan, pelanggan tiba di hotel dan melakukan check-in.
- Jika ada permintaan khusus yang belum dipenuhi, staf hotel akan melakukan pengecekan apakah permintaan tersebut dapat dipenuhi saat itu. Jika tidak, proses akan berakhir di sini.

7. Menginap di Hotel:

- Setelah check-in, pelanggan tinggal di kamar selama waktu yang telah dipesan.

8. Proses Check-out:

- Pada hari check-out, pelanggan melaporkan untuk meninggalkan hotel. Proses check-out dimulai oleh sistem hotel.
- Staf hotel memeriksa kondisi kamar. Jika kondisi kamar baik, deposit (jika ada) dikembalikan kepada pelanggan, dan pelanggan menerima konfirmasi akhir.
- Jika ada kerusakan atau kondisi buruk di kamar, deposit pelanggan akan diambil sebagai biaya tambahan, dan pelanggan diberi tahu.

9. Review dan Selesai:

- Setelah check-out, pelanggan dapat memberikan ulasan terkait pengalaman menginapnya, dan proses reservasi dianggap selesai.

Dalam tahapan proses reservasi dari awal hingga akhir, beberapa masalah utama telah diidentifikasi yang secara signifikan mempengaruhi operasional hotel dan pengalaman pelanggan. Pertama, data menunjukkan tingginya tingkat pembatalan reservasi, yang tercermin dari kolom **is_canceled**. Pembatalan ini sering terjadi setelah reservasi dikonfirmasi, menyebabkan alokasi kamar yang tidak optimal dan kerugian bagi hotel. Hal ini mengindikasikan kurangnya mekanisme penanganan untuk mengurangi pembatalan mendadak, yang perlu dievaluasi lebih lanjut.

Selain itu, terdapat variasi signifikan dalam **lead time**, yaitu waktu antara pemesanan dan kedatangan pelanggan. Variasi ini menciptakan ketidakpastian dalam pengelolaan ketersediaan kamar, yang sering berujung pada *overbooking* atau *underbooking*. Sistem saat ini belum mampu menangani variasi tersebut dengan baik, menyebabkan masalah dalam alokasi kamar yang terkadang berbeda dari yang dipesan (**assigned_room_type** berbeda dengan **reserved_room_type**), terutama pada kondisi *overbooking*.

Selain itu, terdapat kendala dalam penanganan permintaan khusus yang diajukan pelanggan (kolom **total_of_special_requests**). Meskipun sistem menerima permintaan tersebut, banyak permintaan yang tidak diproses secara efektif. Hal ini berdampak pada kepuasan pelanggan, karena layanan tambahan yang diharapkan seringkali tidak tersedia atau tidak sesuai dengan harapan.

Proses pembayaran juga menjadi sumber masalah. Dalam beberapa kasus, pelanggan mengalami kegagalan atau keterlambatan dalam menyelesaikan pembayaran (dilihat dari **adr** dan **reservation_status**). Kegagalan ini tidak hanya membuat pelanggan ragu-ragu untuk melanjutkan reservasi, tetapi juga mengganggu alur check-in dan check-out, yang pada akhirnya menurunkan pengalaman pelanggan secara keseluruhan.

Masalah komunikasi selama proses *check-in* dan *check-out* juga teridentifikasi. Ketiadaan informasi yang tepat waktu dan akurat sering kali menyebabkan kebingungan di antara pelanggan, mengakibatkan keterlambatan dalam *check-in* atau *check-out* dan meningkatkan tingkat ketidakpuasan. Dari temuan ini, diperlukan perbaikan dalam sistem manajemen reservasi, termasuk penyempurnaan komunikasi, pengelolaan permintaan khusus, dan efisiensi proses pembayaran, untuk memastikan operasi hotel berjalan lebih lancar dan pelanggan mendapatkan pengalaman yang lebih baik.

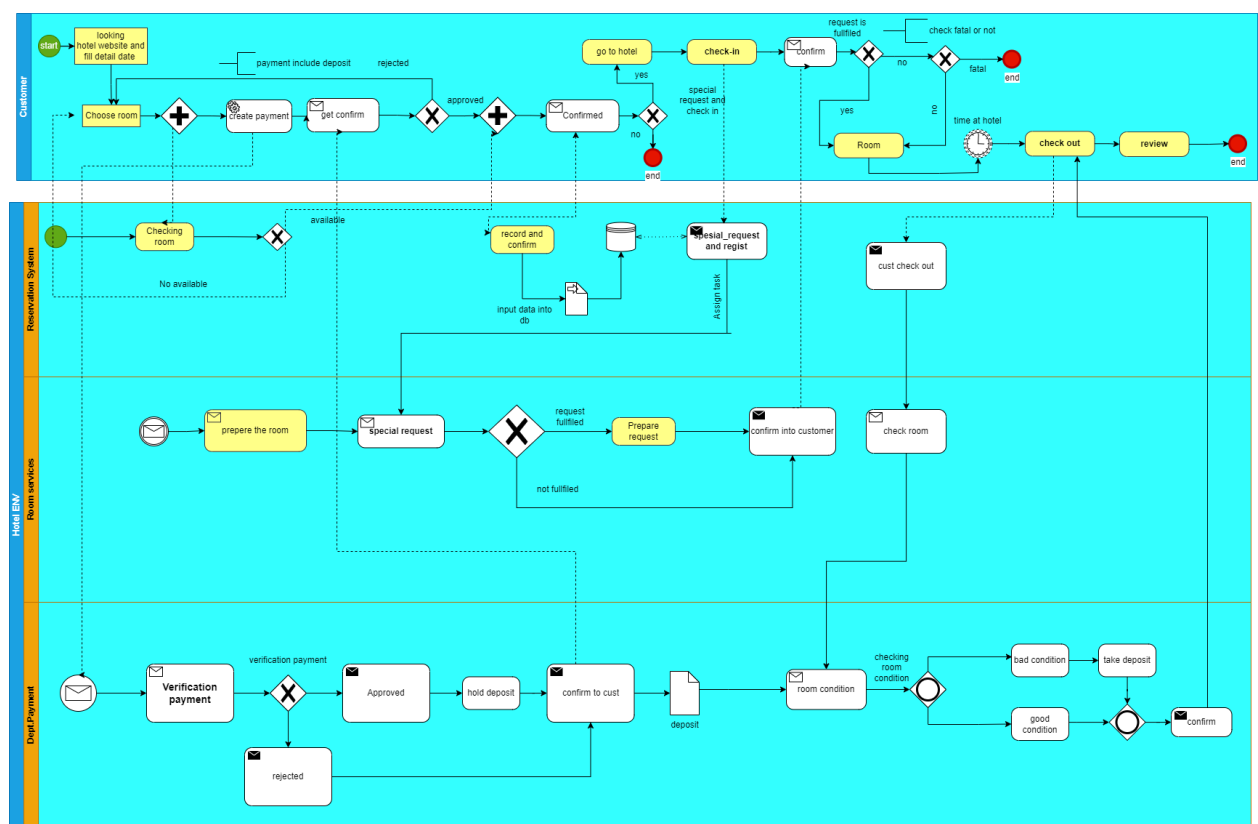
Dari hasil identifikasi masalah yang dijelaskan sebelumnya, sejumlah faktor kunci yang mempengaruhi proses reservasi hotel dapat dirangkum dalam Tabel 1 di bawah ini. Tabel ini memberikan gambaran lebih rinci tentang setiap masalah utama yang telah diidentifikasi, serta kode yang relevan untuk masing-masing faktor. Faktor-faktor seperti tingkat pembatalan reservasi yang tinggi, variabilitas lead time, penanganan permintaan khusus, efisiensi pembayaran, dan komunikasi selama *check-in* dan *check-out*, semuanya memiliki dampak signifikan terhadap kinerja operasional hotel dan tingkat kepuasan pelanggan. Setiap faktor dalam tabel ini akan dianalisis lebih lanjut untuk memahami akar penyebabnya dan bagaimana hal ini dapat diatasi dalam peningkatan sistem manajemen hotel.

Tabel 1. List of Issues in the Reservation Process

Factor	Items	Code
Cancellation	Mengapa tingkat pembatalan reservasi sangat tinggi meskipun kamar sudah dialokasikan?	C1
Lead Time Variability	Bagaimana variabilitas lead time mempengaruhi perencanaan dan ketersediaan kamar hotel?	L1
Special Requests Handling	Mengapa beberapa permintaan khusus pelanggan tidak dapat dipenuhi dengan baik?	SR1
Payment Efficiency	Apa penyebab proses pembayaran sering mengalami keterlambatan atau kegagalan?	P1
Communication Issues	Bagaimana komunikasi selama proses check-in dan check-out dapat mempengaruhi kepuasan pelanggan?	COM1

2.2 Diagram BPMN

Pada bagian ini menjelaskan bagaimana alur standar untuk memodelkan proses bisnis dalam bentuk diagram BPMN (*Business Process Modeling Notation*). Tujuannya adalah untuk membantu memvisualisasikan bagaimana alur kerja atau aktivitas bisnis berjalan dari awal hingga akhir. Dengan menggunakan BPMN, kita dapat dengan mudah memahami dan mendokumentasikan proses-proses yang kompleks, sehingga memudahkan komunikasi antara berbagai tim atau departemen, terutama antara pihak teknis dan non-teknis. Dibawah ini adalah *BPMN* yang memvisualisasikan alur kerja reservasi hotel.



Gambar 1. *BPMN Hotel system*

Proses bisnis reservasi hotel dimulai ketika pelanggan membuka situs hotel dan mengisi detail pemesanan, memilih kamar, serta melakukan pembayaran yang mencakup deposit. Setelah itu, sistem akan melakukan verifikasi pembayaran. Jika pembayaran diterima (*approved*), proses dilanjutkan ke tahap konfirmasi pemesanan. Namun, jika pembayaran ditolak (*rejected*), proses pemesanan akan dihentikan. Setelah pembayaran berhasil, sistem mengkonfirmasi pemesanan dan menyimpannya dalam

database. Ketersediaan kamar akan diperiksa, dan pelanggan akan mendapatkan notifikasi serta diarahkan untuk melakukan *check-in*.

Jika pelanggan memiliki permintaan khusus, hotel akan menerima dan memprosesnya. Jika permintaan tersebut dapat dipenuhi, hotel akan memberikan konfirmasi kepada pelanggan. Namun, jika permintaan tidak dapat dipenuhi, akan ada penanganan lebih lanjut atau eskalasi untuk mencari solusi.

Pada tahap *check-in*, saat pelanggan tiba di hotel, proses *check-in* dilakukan, dan permintaan khusus, jika ada, di registrasi. Apabila terdapat masalah terkait permintaan khusus atau lainnya, eskalasi dapat terjadi yang berpotensi mempengaruhi pengalaman tamu.

Ketika proses *check-out* tiba, hotel akan memeriksa kondisi kamar. Jika kamar dalam kondisi baik, deposit akan dikembalikan kepada pelanggan. Namun, jika ditemukan kerusakan, deposit akan ditahan untuk menutupi biaya kerusakan. Setelah *check-out*, pelanggan akan diminta untuk memberikan ulasan mengenai pengalaman menginap mereka.

Terdapat beberapa permasalahan utama dalam proses bisnis reservasi hotel. Pertama, tingkat pembatalan yang tinggi menjadi tantangan, terutama ketika pembayaran ditolak atau ketika pelanggan membatalkan setelah pemesanan dikonfirmasi, misalnya akibat proses verifikasi yang lambat atau deposit yang tidak sesuai.

Kedua, lead time yang bervariasi, terutama dalam verifikasi kamar dan penanganan permintaan khusus, memperpanjang waktu layanan dan menimbulkan ketidakpastian bagi hotel dalam mengalokasikan sumber daya. Ketiga, permintaan khusus sering tidak dikelola dengan baik, karena diproses secara manual dan memerlukan eskalasi jika tidak terpenuhi, yang dapat menurunkan kepuasan pelanggan.

Keempat, proses pembayaran yang tidak efisien, termasuk verifikasi pembayaran yang sering gagal atau ketidaksesuaian deposit, menimbulkan keraguan bagi pelanggan untuk melanjutkan reservasi.

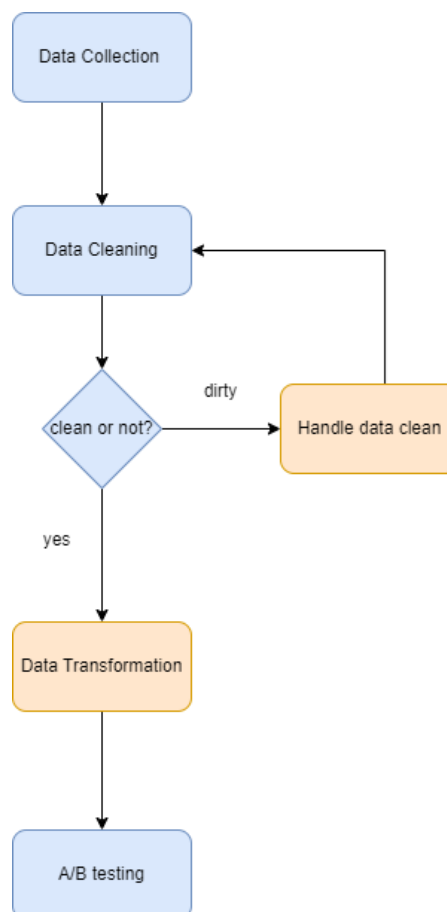
Terakhir, kurangnya komunikasi selama proses *check-in* dan *check-out*, terutama terkait permintaan khusus dan pemeriksaan kondisi kamar, sering menyebabkan kebingungan atau ketidakpuasan pelanggan karena mereka tidak mendapatkan informasi real-time yang dibutuhkan.

BAB III

Data Preparation and Structured Query Language

3.1 Data Preparation

Bagan dibawah ini adalah alur proses data preparation, data preparation membantu memastikan bahwa data yang digunakan bersih, relevan, dan dalam format yang sesuai untuk analisis lebih lanjut. Secara umum, ada beberapa langkah utama yang terlibat dalam data preparation, yang dapat kita visualisasikan dalam bentuk flowchart. Flowchart ini akan memandu kita melalui tahapan seperti pengumpulan data, pembersihan data, transformasi data, hingga data testing. Dapat kita lihat pada Gambar 2 langkah-langkah tersebut secara lebih rinci.



Gambar 2. *Flow data preparation.*

Dibawah ini adalah potongan kode untuk melakukan data cleansing data kami menggunakan beberapa langkah untuk melakukan cleansing data yakni :

1. Melakukan checking terhadap data duplikat
2. Menangani data yang tidak konsisten.
3. Melakukan checking dan imputasi untuk data yang missing atau data hilang.
4. Melakukan analisis outliers

Kode Program Python 1. Proses *data cleansing dan handle missing value* pada Python.

```
1. # 1. Menghapus duplikasi
2. duplikat = data.duplicated().sum()
3. print("Jumlah duplikat data: ", duplikat)
4.
5. data.drop_duplicates(inplace=True)
6.
7. # 2. Menangani Data yang Tidak Konsisten
8.
9. # Misalnya, kolom 'meal' mungkin memiliki variasi seperti 'BB', 'FB', 'HB', dll.
10. # Menghitung modus dari kolom 'meal'
11. meal_mode = data['meal'].mode()[0]
12. meal_mode
13. # Mengganti nilai 'Undefined' dan 'Undef' menjadi modus
14. # data['meal'] = data['meal'].replace(['Undefined', 'Undef'], meal_mode)
15. data.loc[data['meal'].isin(['Undefined', 'Undef']), 'meal'] = meal_mode
16.
17.
18. # Lakukan pengecekan pada kolom-kolom kategori lainnya juga
19. data['market_segment'] = data['market_segment'].str.strip() # Menghapus spasi yang
    tidak perlu
20. data['distribution_channel'] = data['distribution_channel'].str.strip()
21.
22. # 3. Menangani Data yang Hilang
23. # Mengisi nilai yang hilang dengan strategi yang tepat
24. # Untuk kolom numerik, kita bisa menggunakan mean atau median
25. data['lead_time'] = data['lead_time'].fillna(data['lead_time'].mean())
26. data['stays_in_weekend_nights'] =
    data['stays_in_weekend_nights'].fillna(data['stays_in_weekend_nights'].mean())
27. data['adults'] = data['adults'].fillna(data['adults'].median())
28. data['children'] = data['children'].fillna(data['children'].median())
29. data['adr'] = data['adr'].fillna(data['adr'].mean())
30. data['total_of_special_requests'] =
    data['total_of_special_requests'].fillna(data['total_of_special_requests'].median())
31.
32. # Kolom agent dan company dapat diisi dengan '0' karena bernilai numerik, atau None
    jika tidak ada data.
33. data['agent'] = data['agent'].fillna(0)
34. data['company'] = data['company'].fillna(0)
35.
36. # Untuk kolom 'country', kita bisa mengisi dengan mode (nilai yang paling sering
    muncul)
37. data['country'] = data['country'].fillna(data['country'].mode()[0])
38.
39. # Jika kolom 'company' terlalu banyak missing value dan dianggap tidak relevan, kita
    bisa menghapusnya
40. data.drop(columns=['company'])
```

Berikut adalah penjelasan untuk proses data cleansing yang kami lakukan sebelum melakukan tahap selanjutnya :

- **Meng-check Duplikasi Data:** Data yang memiliki nilai duplikat sering kali mengindikasikan pencatatan ganda yang tidak perlu. Pada langkah ini, kami menghitung jumlah baris duplikat menggunakan `data.duplicated().sum()`. Jika ditemukan duplikat, kita menghapusnya menggunakan `data.drop_duplicates(inplace=True)`, yang memastikan setiap baris dalam dataset adalah unik.
- **Menangani Data yang Tidak Konsisten:** Ketidakkonsistenan dalam data kategori dapat mengurangi akurasi analisis. Misalnya, dalam kolom `meal`, ditemukan nilai seperti "Undefined" dan "Undef". Nilai-nilai ini diubah menjadi modus atau nilai yang paling sering muncul dalam kolom tersebut, yang diperoleh dari `data['meal'].mode()[0]`. Dengan cara ini, kita memastikan konsistensi data kategori. Selain itu, kita juga menghapus spasi berlebih pada kolom `market_segment` dan `distribution_channel` untuk mengurangi ketidakcocokan data.
- **Menangani Data yang Hilang:** Data yang hilang ditangani berdasarkan jenis dan kegunaan kolomnya:
 - **Kolom Numerik:** Untuk kolom numerik seperti `lead_time`, `stays_in_weekend_nights`, `adults`, `children`, `adr`, dan `total_of_special_requests`, nilai yang hilang diisi menggunakan nilai rata-rata atau median dari kolom tersebut, tergantung pada distribusi data. Pengisian dengan rata-rata atau median ini membantu menjaga nilai sentral dari data.
 - **Kolom Kategori:** Pada kolom `country`, nilai yang hilang diisi dengan modus, yang merupakan nilai yang paling sering muncul. Hal ini dilakukan agar nilai yang ditambahkan tetap sejalan dengan distribusi umum kolom tersebut.
 - **Kolom dengan Missing Value Tinggi:** Pada kolom `agent` dan `company`, jika nilai hilang terlalu banyak, nilainya diisi dengan 0. Selain itu, kolom `company` yang mungkin memiliki banyak data yang hilang dan dianggap kurang relevan untuk analisis ini dapat dihapus.

Langkah-langkah ini bertujuan untuk meningkatkan kualitas data dengan menghapus duplikasi, mengatasi ketidakkonsistenan, dan menangani data yang hilang sehingga analisis dapat dilakukan dengan lebih akurat.

Melakukan analisis outliers

Kode Program Python 2. Process check data outliers

```
1. # Langkah 2: Menghitung IQR untuk kolom tertentu (misalnya 'lead_time')
2. Q1 = data['lead_time'].quantile(0.25)
3. Q3 = data['lead_time'].quantile(0.75)
4. IQR = Q3 - Q1
5.
6. # Penyesuaian batas bawah dan atas dengan faktor pengali yang lebih besar
7. factor = 3 # Anda bisa coba dengan 2.5 atau 3
8. lower_bound = Q1 - factor * IQR
9. upper_bound = Q3 + factor * IQR
10.
11. # Mendeteksi outlier
12. outliers = data[(data['lead_time'] < lower_bound) | (data['lead_time'] >
    upper_bound)]
13.
14. # Winsorizing - Mengganti nilai outliers dengan batas bawah dan atas
15. data['lead_time'] = data['lead_time'].apply(lambda x: lower_bound if x < lower_bound
    else upper_bound if x > upper_bound else x)
16.
17. # Menampilkan hasil
18. print("Jumlah Outliers setelah Winsorizing:", len(outliers))
19. print("Data setelah Winsorizing:")
20. print(data['lead_time'].describe())
21.
```

Dalam proses analisis data ini, langkah pertama adalah menghitung Interquartile Range (IQR) untuk kolom `lead_time`, dengan tujuan untuk mendeteksi dan menangani nilai outliers yang berpotensi memengaruhi distribusi data secara signifikan. IQR dihitung sebagai selisih antara kuartil ketiga (Q3) dan kuartil pertama (Q1). Berdasarkan hasil perhitungan, kami menetapkan batas bawah dan batas atas dengan menggunakan faktor 3 untuk faktor pengali, yang akan memberikan jangkauan lebih luas dibandingkan faktor standar 1.5 atau 2.5. Penyesuaian ini dilakukan untuk menyaring nilai ekstrem dengan lebih toleran, namun tetap menjaga integritas data.

Setelah batas ditentukan, kami mendeteksi outliers yang berada di luar batas tersebut dan melakukan Winsorizing. Teknik Winsorizing ini menggantikan nilai outliers dengan batas bawah atau batas atas yang ditentukan. Misalnya, nilai yang berada di bawah batas bawah diganti dengan nilai batas bawah, sedangkan nilai yang berada di atas batas atas diganti dengan nilai batas atas. Pendekatan ini dipilih karena tidak menghilangkan data, namun hanya menyesuaikan nilainya. Sehingga, metode ini lebih aman dalam menjaga distribusi data tanpa merusak integritas dataset. Winsorizing mengurangi risiko distorsi pada analisis data yang sering disebabkan oleh nilai ekstrem, namun tetap menjaga kontribusi nilai data secara keseluruhan.

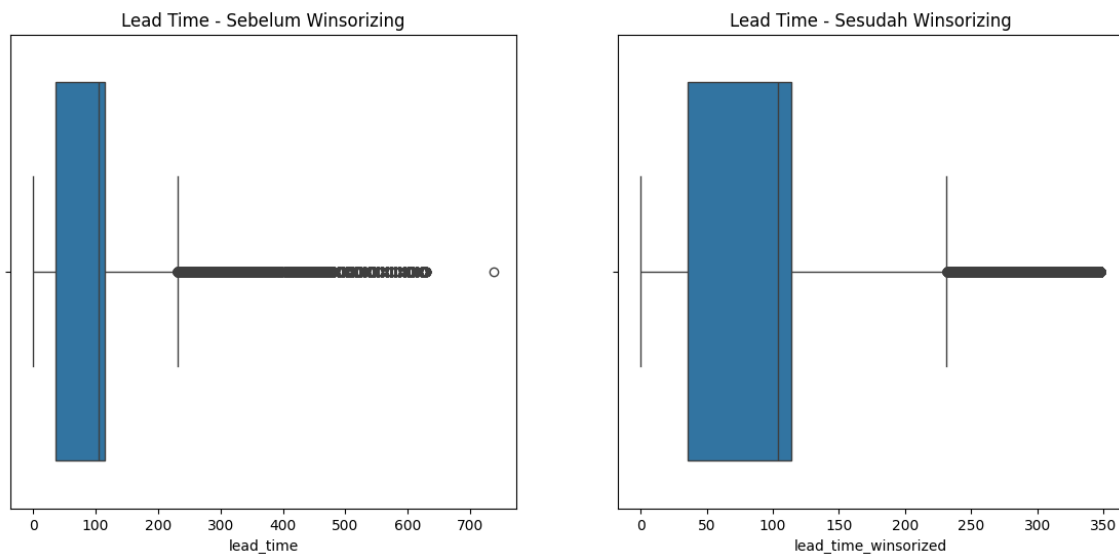
Kode Program Python 3. Visualisasi data sesudah dan sebelum handling outliers

```

1. comparison_df = data.copy()
2. comparison_df['lead_time_winsorized'] = winsorizing
3.
4. # Membuat boxplot sebelum Winsorizing
5. plt.figure(figsize=(14, 6))
6.
7. plt.subplot(1, 2, 1)
8. sns.boxplot(data=data, x='lead_time') # original_data adalah data sebelum
Winsorizing
9. plt.title("Lead Time - Sebelum Winsorizing")
10.
11. # Membuat boxplot sesudah Winsorizing
12. plt.subplot(1, 2, 2)
13. sns.boxplot(x=comparison_df['lead_time_winsorized']) # data setelah Winsorizing
14. plt.title("Lead Time - Sesudah Winsorizing")
15.
16. plt.show()
17.
18. plt.show()
19. print("Jumlah Outliers setelah Winsorizing:", len(outliers))
20. print("Data setelah Winsorizing:")
21. print(data['lead_time'].describe())
22.

```

Visualisasi hasil data yang sudah di handling dengan Winsorizing dan yang belum:



3.2 Data Extraction

Tujuan dari data extraction adalah untuk mengumpulkan informasi yang relevan dan menyusunnya dalam bentuk yang lebih mudah diolah, dianalisis, atau dimigrasikan ke sistem lain. Ini sering menjadi bagian dari proses ETL (Extract, Transform, Load), di mana setelah data diekstraksi, data kemudian diubah (transform) agar sesuai dengan format yang diinginkan dan dimuat (load) ke dalam sistem target.

Dalam bagian data extraction kami memiliki fokus analisa yaitu untuk mengevaluasi kinerja hotel dan memahami pola reservasi pelanggan, berikut ini adalah langkah-langkah untuk mendapatkan fokus analisa kami :

3.1.1 Menghitung Tingkat Pembatalan Hotel untuk Setiap Jenis Hotel Berdasarkan Tahun.

Query ini bertujuan untuk menghitung tingkat pembatalan reservasi di hotel berdasarkan jenis hotel dan tahun kedatangan. Dengan query ini kita mendapatkan wawasan mengenai tren pembatalan hotel dari tahun ketahun.

Kode Program SQL 4. Penggunaan Query CTE.

```
1. with persen as (
2.   SELECT
3.     hotel,
4.     arrival_date_year,
5.     ROUND(100.0 * SUM(is_canceled) / COUNT(*), 2) AS cancellation_rate
6.   FROM
7.     clean_midterm_hotel_data cmhd
8.   GROUP BY
9.     hotel,
10.    arrival_date_year
11. )
12. select * from persen
13.
```

Berikut adalah output dari query diatas :

	A-z hotel	123 arrival_date_year	123 cancellation_rate
1	City Hotel	2,015	43.88
2	City Hotel	2,016	40.4
3	City Hotel	2,017	42.5
4	Resort Hotel	2,015	25.72
5	Resort Hotel	2,016	26.55
6	Resort Hotel	2,017	30.76

Berdasarkan hasil analisis, dapat disimpulkan bahwa tingkat pembatalan reservasi di City Hotel dan Resort Hotel menunjukkan tren yang berbeda selama periode 2015 hingga 2017. **City Hotel** secara konsisten memiliki tingkat pembatalan yang lebih tinggi, dengan persentase yang berkisar antara 40.4% hingga 43.88%. Meskipun terjadi sedikit penurunan pada tahun 2016, tingkat pembatalan kembali naik pada tahun 2017. Di sisi lain, **Resort Hotel** menunjukkan tren peningkatan yang lebih signifikan. Pada tahun 2015, tingkat pembatalan sebesar 25.72%, namun meningkat menjadi 30.76% di tahun 2017. Meskipun tingkat pembatalan di Resort Hotel masih lebih rendah dibandingkan City Hotel, peningkatan yang terus-menerus ini perlu menjadi perhatian.

3.1.2 Rata-rata *lead_time* Berdasarkan *total_of_special_request*

Query ini bertujuan untuk melihat hubungan antara rata-rata *lead_time* atau waktu kedatangan pelanggan dengan *total_of_special_request* atau permintaan spesial, analisis ini dapat memberikan insight pola customer behavior terhadap jarak waktu *check-in*.

Kode Program SQL 5. Penggunaan Query CTE.

```
1. WITH avg_lead_time_overall AS (
2.     SELECT AVG(lead_time) AS overall_avg_lead_time
3.     FROM clean_midterm_hotel_data
4. )
5. SELECT
6.     total_of_special_requests,
7.     ROUND(AVG(lead_time), 2) AS average_lead_time
8. FROM
9.     clean_midterm_hotel_data, avg_lead_time_overall
10. WHERE
11.     lead_time > avg_lead_time_overall.overall_avg_lead_time
12. GROUP BY
13.     total_of_special_requests
14. ORDER BY
15.     total_of_special_requests;
```

Output dari query di atas adalah :

	123 total_of_special_requests	123 average_lead_time
1	0	156.47
2	1	143.52
3	2	141.99
4	3	142.3
5	4	152.37
6	5	145.32

Berdasarkan output yang ditampilkan, terdapat pola hubungan antara jumlah permintaan khusus (*total_of_special_requests*) dan rata-rata lead time (*average_lead_time*). Terlihat bahwa pelanggan yang tidak mengajukan permintaan khusus (0 permintaan) memiliki lead time tertinggi sebesar 156.47 hari, yang menunjukkan bahwa mereka cenderung memesan jauh-jauh hari tanpa kebutuhan tambahan. Di sisi lain, rata-rata lead time menurun ketika jumlah permintaan khusus bertambah hingga 2 permintaan, dengan lead time terendah sebesar 141.99 hari. Hal ini bisa mengindikasikan bahwa pelanggan dengan sedikit permintaan khusus mungkin memesan lebih dekat ke waktu menginap, mungkin karena keinginan untuk memastikan ketersediaan fasilitas tertentu.

Menariknya, ketika jumlah permintaan khusus bertambah menjadi 3 hingga 5 permintaan, rata-rata lead time sedikit meningkat kembali, dengan puncaknya

pada 152.37 hari untuk 4 permintaan. Hal ini dapat menunjukkan bahwa pelanggan dengan banyak kebutuhan khusus cenderung memesan lebih awal untuk memastikan layanan dan fasilitas spesifik dapat terpenuhi oleh hotel. Secara keseluruhan, data ini menggambarkan bahwa jumlah permintaan khusus memengaruhi waktu pemesanan, di mana pelanggan dengan kebutuhan sedikit atau tanpa permintaan cenderung memesan lebih cepat, sementara pelanggan dengan permintaan lebih banyak mungkin mengambil langkah lebih awal untuk menghindari ketidakpastian dalam ketersediaan layanan.

3.1.3 Menghitung Pendapatan Tertinggi berdasarkan Tanggal Reservasi

Pada tahap ini kami akan menghitung pendapatan tertinggi berdasarkan tanggal reservasi, analisis ini bertujuan untuk melihat pola waktu dari customer.

Kode Program SQL 6. Penggunaan Query

```
1. SELECT
2.     reservation_status_date,
3.     ROUND(AVG(adr), 2) AS total_revenue
4. FROM
5.     midterm_hotel_data
6. WHERE
7.     reservation_status = 'Check-Out'
8.     AND total_of_special_requests > 2
9. GROUP BY
10.    reservation_status_date
11. ORDER BY
12.    total_revenue DESC
13. LIMIT 1;
```

Output dari query diatas :

	A-2 reservation_status_date	123 total_revenue
1	2017-01-02	287.7

Pada 2 Januari 2017 adalah tanggal dengan total pendapatan tertinggi untuk periode yang dianalisis. Hal ini bisa jadi terkait dengan peningkatan permintaan selama periode liburan Tahun Baru, ketika banyak orang bepergian dan menginap di hotel.

Pendapatan yang tinggi ini menunjukkan bahwa hotel berhasil memaksimalkan pendapatan per reservasi pada tanggal tersebut, mungkin karena harga yang lebih tinggi selama musim ramai atau tingkat hunian yang tinggi.

Jika analisis ini dilakukan terhadap rentang waktu yang lebih panjang, seperti seluruh tahun, maka kita dapat melihat apakah pola ini konsisten selama periode liburan atau apakah hanya terjadi pada tanggal-tanggal tertentu saja.

3.1.4 Hubungan Antara Lead Time dan Keberhasilan Pembayaran

Query ini bertujuan untuk menganalisis hubungan antara *lead time* dan keberhasilan pembayaran (diukur dengan nilai *adr*) menggunakan *Common Table Expression* (CTE). Kami akan mengelompokkan data berdasarkan rentang *lead_time* dan menghitung rata-rata *adr* untuk setiap kelompok.

Kode Program SQL 7. Penggunaan Query

```
1. WITH lead_time_groups AS (
2.     SELECT
3.         CASE
4.             WHEN lead_time <= 30 THEN '0-30'
5.             WHEN lead_time BETWEEN 31 AND 90 THEN '31-90'
6.             WHEN lead_time BETWEEN 91 AND 180 THEN '91-180'
7.             ELSE '181+'
8.         END AS lead_time_range,
9.         adr
10.    FROM
11.        clean_midterm_hotel_data
12. )
13. SELECT
14.     lead_time_range,
15.     ROUND(AVG(adr), 2) AS average_adr
16. FROM
17.     lead_time_groups
18. GROUP BY
19.     lead_time_range
20. ORDER BY
21.     lead_time_range;
```

Berikut hasil output query di atas:

	A-Z lead_time_range	123 average_adr
1	0-30	99.76
2	181+	95.67
3	31-90	105.3
4	91-180	103.85

Berdasarkan output yang diberikan, terlihat bahwa terdapat pola hubungan antara *lead time* (waktu antara pemesanan dan check-in) dengan rata-rata tarif harian (ADR). Pemesanan dengan *lead time* sedang (31–90 hari) mencatat ADR tertinggi sebesar 105.3, diikuti oleh rentang 91–180 hari dengan rata-rata ADR 103.85. Hal ini menunjukkan bahwa pemesanan dalam rentang waktu tersebut

cenderung lebih mahal, kemungkinan karena masih berada dalam periode permintaan tinggi tanpa diskon besar.

Sementara itu, lead time singkat (0–30 hari) memiliki ADR lebih rendah sebesar 99.76, yang dapat mengindikasikan adanya diskon last-minute untuk menarik pelanggan dan mengisi kamar yang tersisa. Di sisi lain, pemesanan dengan lead time sangat panjang (lebih dari 181 hari) mencatat ADR terendah sebesar 95.67, yang kemungkinan besar disebabkan oleh promo early bird untuk mendorong pemesanan jauh-jauh hari.

Secara keseluruhan, data ini menunjukkan bahwa hotel menerapkan strategi harga dinamis, di mana tarif lebih tinggi dikenakan untuk pemesanan di rentang waktu sedang, sementara pemesanan yang dilakukan terlalu awal atau mendekati hari check-in diberi tarif lebih rendah untuk memaksimalkan okupansi dan pendapatan.

BAB IV

Python Programming

4.1 Exploratory Data Analysis

EDA atau Exploratory Data Analysis adalah proses penting dalam analisis data. Dan kami akan menjelaskan keadaan data yang kami analisis berdasarkan grafik. Ini adalah langkah awal yang memungkinkan kita untuk menjelajahi dan memahami data yang dimiliki, atau secara sederhana EDA adalah proses uji investigasi yang bertujuan mengidentifikasi pola, menemukan anomali data, dan menguji hipotesis. Salah satu langkah yang diterapkan dalam proses EDA adalah membuat daftar pertanyaan yang berkaitan dengan analisis data, pada analisis ini kami memiliki beberapa pertanyaan yang dibagi menjadi beberapa sub-bab.

4.1.1 Rasio antara *lead_time* dengan Jumlah Reservasi yang Dibatalkan

Dalam permasalahan ini kami ingin mengetahui bagaimana rasio antara *lead_time* dengan jumlah reservasi yang dibatalkan, ini merupakan analisis awal agar dapat menghasilkan insight yang bermanfaat. Berikut adalah kode pemrograman untuk melihat rasio *lead_time* dengan reservasi yang batal

Kode Program SQL 8. Menghitung rasio *lead_time*

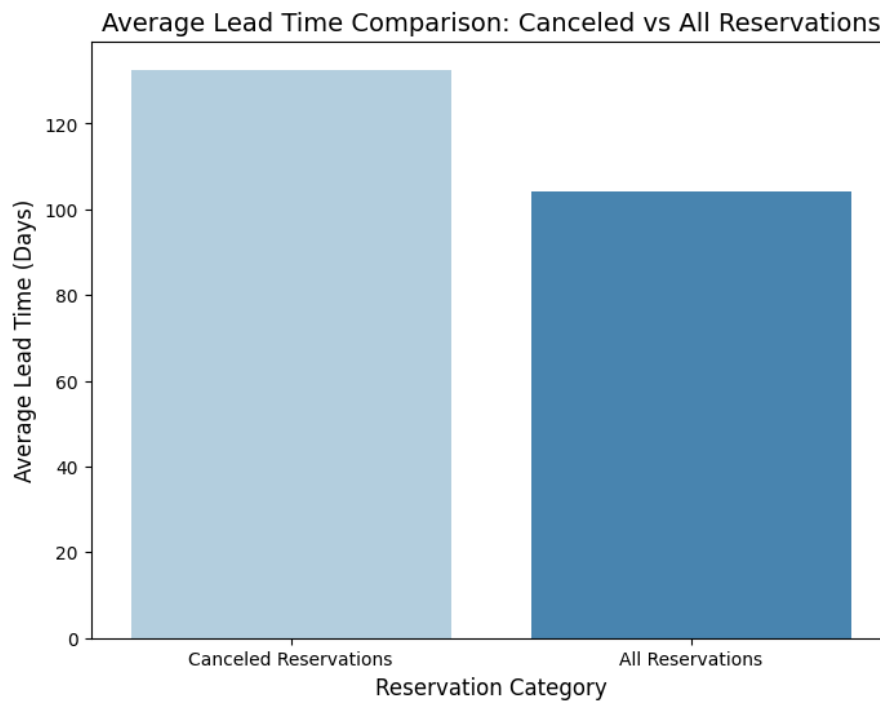
```

1. # 1. Filter data untuk reservasi yang dibatalkan (is_canceled == 1)
2. canceled_reservations = data[data['is_canceled'] == 1]
3.
4. # 2. Menghitung rata-rata lead_time untuk reservasi yang dibatalkan
5. avg_lead_time_canceled = canceled_reservations['lead_time'].mean()
6.
7. # 3. Menghitung rata-rata lead_time untuk semua reservasi
8. avg_lead_time_all = data['lead_time'].mean()
9.
10. # 4. Menyimpan nilai rata-rata dalam sebuah DataFrame untuk visualisasi
11. avg_lead_time_data = pd.DataFrame({
12.     'Category': ['Canceled Reservations', 'All Reservations'],
13.     'Lead Time': [avg_lead_time_canceled, avg_lead_time_all]
14. })
15.
16. # 5. Membuat bar plot untuk membandingkan rata-rata lead_time
17. plt.figure(figsize=(8, 6))
18. sns.barplot(x='Category', y='Lead Time', data=avg_lead_time_data,
19.             palette='Blues', hue='Category')
20.
21. # Menambahkan title dan labels
22. plt.title('Average Lead Time Comparison: Canceled vs All Reservations',
23.          fontsize=14)
24. plt.ylabel('Average Lead Time (Days)', fontsize=12)
25. plt.xlabel('Reservation Category', fontsize=12)
26. # Menampilkan grafik
27. plt.show()
28.
29. # Menampilkan hasil rata-rata

```

```
30. print(f"Rata-rata lead_time untuk reservasi yang dibatalkan:
    {avg_lead_time_canceled}")
31. print(f"Rata-rata lead_time untuk semua reservasi: {avg_lead_time_all}")
```

Output dari kode diatas menghasilkan :



```
Rata-rata lead_time untuk reservasi yang dibatalkan: 129.06013721189453
Rata-rata lead_time untuk semua reservasi: 102.36051356299284
```

Berdasarkan grafik, terlihat bahwa reservasi yang dibatalkan memiliki rata-rata lead time sebesar 129,06 hari, lebih tinggi dibandingkan dengan rata-rata lead time dari seluruh reservasi yang hanya 102,36 hari. Pola ini menunjukkan bahwa semakin panjang lead time (waktu antara pemesanan dan check-in), semakin besar kemungkinan reservasi tersebut dibatalkan. Hal ini mungkin disebabkan oleh ketidakpastian rencana pelanggan yang memesan jauh-jauh hari, seperti perubahan jadwal, kebutuhan mendesak, atau penawaran lebih baik dari hotel lain. Sebaliknya, reservasi yang dilakukan dengan lead time lebih pendek umumnya memiliki risiko pembatalan lebih rendah, karena keputusan menginap dibuat lebih dekat dengan waktu check-in, sehingga peluang terjadinya perubahan rencana lebih kecil. Temuan ini mengindikasikan bahwa hotel dapat mempertimbangkan strategi mitigasi risiko, seperti pembayaran di muka, penalti pembatalan, atau insentif tambahan untuk memastikan tamu tetap melanjutkan reservasi mereka.

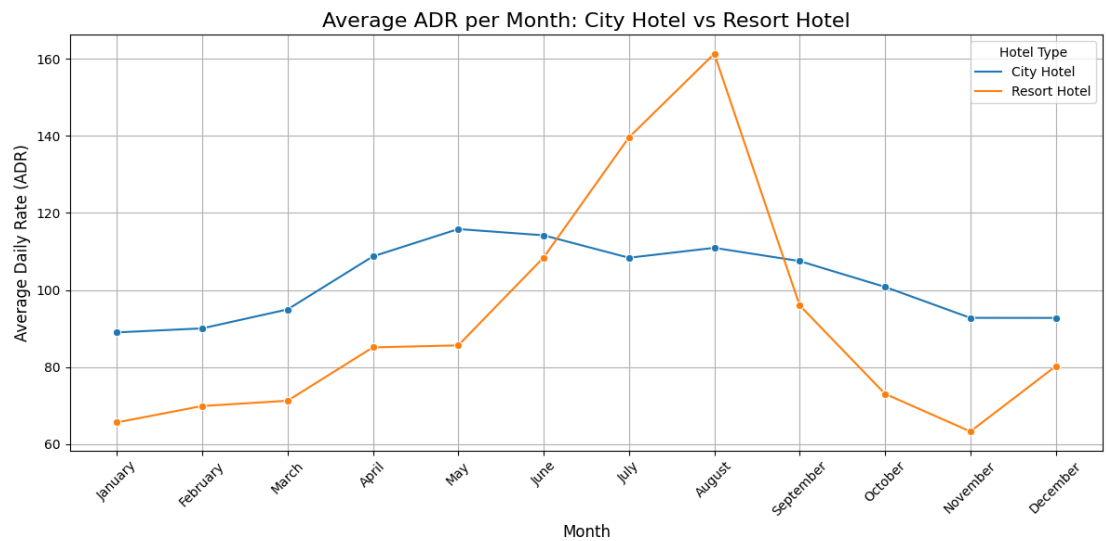
4.1.2 Pola Pendapatan Hotel Selama beberapa Bulan Tertentu

Analisis pola pendapatan hotel selama beberapa bulan tertentu dilakukan untuk memahami perbedaan tren **Average Daily Rate (ADR)** antara **City Hotel** dan **Resort Hotel** sepanjang tahun. Pada tahap ini, data dikelompokkan berdasarkan jenis hotel dan bulan kedatangan tamu untuk menghitung rata-rata ADR setiap bulan. Dengan menyusun bulan dalam urutan kalender, kita dapat mengidentifikasi kapan periode puncak atau penurunan pendapatan terjadi pada masing-masing jenis hotel. Visualisasi dalam bentuk **line plot** digunakan untuk memudahkan perbandingan dan interpretasi pola perubahan ADR antara kedua tipe hotel, yang dapat memberikan wawasan mengenai dampak musiman dan perilaku tamu dalam memilih akomodasi.

Kode Program SQL 9. Menghitung pola pendapatan hotel selama beberapa bulan tertentu

```
1. # Pastikan kolom 'arrival_date_month' dalam urutan yang benar
2. month_order = ['January', 'February', 'March', 'April', 'May', 'June',
3.               'July', 'August', 'September', 'October', 'November',
4.               'December']
5. # 1. Kelompokkan data berdasarkan bulan dan jenis hotel, lalu hitung rata-
   rata ADR per bulan
6. avg_adr_per_month =
   data.groupby(['hotel', 'arrival_date_month'])['adr'].mean().reset_index()
7.
8. # Pastikan bulan diurutkan sesuai dengan urutan bulan kalender
9. avg_adr_per_month['arrival_date_month'] =
   pd.Categorical(avg_adr_per_month['arrival_date_month'],
10.               categories=month_order, ordered=True)
11. avg_adr_per_month = avg_adr_per_month.sort_values('arrival_date_month')
12. # 2. Membuat line plot untuk membandingkan pola pendapatan (ADR) antara
   Resort Hotel dan City Hotel
13. plt.figure(figsize=(12, 6))
14. sns.lineplot(x='arrival_date_month', y='adr', hue='hotel',
15.               data=avg_adr_per_month, marker='o')
16. # Menambahkan title dan labels
17. plt.title('Average ADR per Month: City Hotel vs Resort Hotel', fontsize=16)
18. plt.xlabel('Month', fontsize=12)
19. plt.ylabel('Average Daily Rate (ADR)', fontsize=12)
20. plt.xticks(rotation=45) # Rotate x-axis labels for better readability
21. plt.legend(title='Hotel Type')
22. plt.grid(True)
23.
24. # Menampilkan grafik
25. plt.tight_layout()
26. plt.show()
```

Output dari kode diatas adalah sebagai berikut :



Grafik di atas menunjukkan perbandingan rata-rata tarif harian (ADR) antara **City Hotel** dan **Resort Hotel** berdasarkan bulan. Dari grafik tersebut, terlihat bahwa **City Hotel** memiliki pola yang lebih stabil sepanjang tahun. Tarif harian di City Hotel secara perlahan meningkat dari bulan Januari hingga mencapai puncaknya di bulan Mei dan Juni dengan ADR sekitar 120. Setelah itu, tarif sedikit menurun namun tetap stabil hingga Desember di kisaran antara 100 hingga 120.

Berbeda dengan **Resort Hotel**, yang menunjukkan variasi tarif yang jauh lebih besar. Di awal tahun, ADR Resort Hotel berada di kisaran 60 hingga 80. Namun, mulai bulan Juni, terdapat kenaikan yang sangat signifikan hingga mencapai puncaknya di bulan Agustus dengan ADR sekitar 160. Setelah bulan Agustus, ADR menurun tajam, terutama pada bulan September hingga Desember, di mana tarif kembali mendekati level awal tahun.

Secara keseluruhan, dapat disimpulkan bahwa **Resort Hotel** mengalami fluktuasi tarif yang lebih signifikan dengan puncak pada musim liburan musim panas, sementara **City Hotel** cenderung mempertahankan tarif yang lebih stabil sepanjang tahun tanpa adanya perubahan besar yang terkait dengan musim.

4.1.3 Distribusi *canceled reservation* berdasarkan negara asal turis.

Analisis distribusi pembatalan reservasi berdasarkan negara asal turis bertujuan untuk mengidentifikasi negara-negara dengan frekuensi pembatalan tertinggi pada **City Hotel** dan **Resort Hotel**. Pada tahap ini, data difilter untuk menampilkan hanya reservasi yang dibatalkan, kemudian dilakukan pengelompokan berdasarkan negara dan jenis hotel untuk menghitung jumlah pembatalan. Selanjutnya, dipilih 10 negara teratas dengan jumlah pembatalan terbanyak untuk setiap kategori hotel. Visualisasi dalam bentuk **bar plot** digunakan untuk memperjelas perbandingan antar negara dan memberikan wawasan mengenai pola pembatalan dari turis internasional di kedua tipe hotel. Hasil ini diharapkan dapat membantu manajemen hotel dalam memahami karakteristik turis dari berbagai negara terkait pembatalan reservasi.

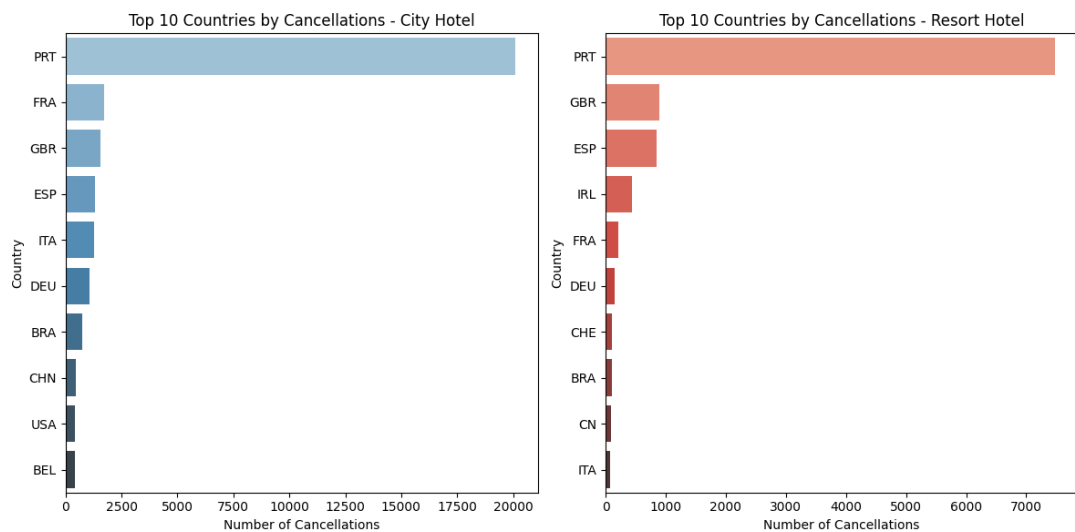
Kode Program SQL 10. Menghitung distribusi *canceled reservation* berdasarkan negara asal turis

```

1. # 1. Filter data untuk reservasi yang dibatalkan (is_canceled == 1)
2. canceled_reservations = data[data['is_canceled'] == 1]
3.
4. # 2. Menghitung jumlah pembatalan berdasarkan negara dan jenis hotel
5. canceled_by_country = canceled_reservations.groupby(['hotel',
6. 'country']).size().reset_index(name='cancellation_count')
7.
8. # 3. Memilih top 10 negara dengan pembatalan terbanyak untuk setiap hotel
9. top_canceled_city = canceled_by_country[canceled_by_country['hotel'] ==
10. 'City Hotel'].nlargest(10, 'cancellation_count')
11. top_canceled_resort = canceled_by_country[canceled_by_country['hotel'] ==
12. 'Resort Hotel'].nlargest(10, 'cancellation_count')
13.
14. # 4. Visualisasi jumlah pembatalan berdasarkan negara untuk City Hotel
15. plt.figure(figsize=(12, 6))
16.
17. plt.subplot(1, 2, 1) # Plot pertama untuk City Hotel
18. sns.barplot(x='cancellation_count', y='country', data=top_canceled_city,
19. palette='Blues_d')
20. plt.title('Top 10 Countries by Cancellations - City Hotel')
21. plt.xlabel('Number of Cancellations')
22. plt.ylabel('Country')
23.
24. # 5. Visualisasi jumlah pembatalan berdasarkan negara untuk Resort Hotel
25. plt.subplot(1, 2, 2) # Plot kedua untuk Resort Hotel
26. sns.barplot(x='cancellation_count', y='country', data=top_canceled_resort,
27. palette='Reds_d')
28. plt.title('Top 10 Countries by Cancellations - Resort Hotel')
29. plt.xlabel('Number of Cancellations')
30. plt.ylabel('Country')
31.
32. # Menampilkan grafik
33. plt.tight_layout()
34. plt.show()

```


Visualisasi untuk jumlah negara asal turis yang melakukan cancel tertinggi adalah :



Grafik tersebut menampilkan 10 negara teratas dengan jumlah pembatalan tertinggi untuk **City Hotel** dan **Resort Hotel**. Pada **City Hotel**, negara dengan pembatalan terbanyak adalah Portugal (PRT), dengan jumlah yang sangat signifikan mendekati 20.000 pembatalan, jauh melampaui negara-negara lain. Di urutan berikutnya, terdapat Prancis (FRA), Inggris (GBR), Spanyol (ESP), dan Italia (ITA) dengan jumlah pembatalan berkisar antara 2.000 hingga 3.000. Negara-negara lainnya, seperti Jerman (DEU), Brasil (BRA), China (CHN), Amerika Serikat (USA), dan Belgia (BEL) juga masuk dalam daftar 10 besar, namun dengan angka yang lebih rendah.

Sementara itu, pada **Resort Hotel**, Portugal (PRT) kembali memimpin dengan lebih dari 7.000 pembatalan. Inggris (GBR) dan Spanyol (ESP) mengikuti di posisi kedua dan ketiga dengan masing-masing sekitar 3.000 dan 2.500 pembatalan. Negara-negara seperti Irlandia (IRL), Prancis (FRA), Jerman (DEU), dan Swiss (CHE) juga berada di daftar 10 besar, namun dengan jumlah yang lebih rendah dibandingkan Portugal.

Secara keseluruhan, Portugal mendominasi jumlah pembatalan di kedua jenis hotel, menunjukkan bahwa pelanggan dari Portugal memiliki kecenderungan membatalkan reservasi lebih banyak. **City Hotel** juga mencatatkan lebih banyak pembatalan dibandingkan **Resort Hotel**, yang mengindikasikan kemungkinan

perbedaan dalam preferensi dan pola pembatalan antara kedua tipe hotel tersebut. Negara-negara Eropa lainnya seperti Prancis, Inggris, Spanyol, dan Jerman juga menunjukkan jumlah pembatalan yang cukup tinggi di kedua jenis hotel.

4.1.4 User Behavior di Last Minute Sebelum Check-In

Analisis ini berfokus pada perilaku pemesanan **last-minute**, yaitu reservasi dengan **lead time kurang dari 7 hari** sebelum tanggal check-in. Pemesanan last-minute memiliki karakteristik dan risiko tertentu, seperti potensi pembatalan yang lebih tinggi. Analisis ini bertujuan untuk melihat seberapa sering pemesanan last-minute terjadi di **Resort Hotel** dan **City Hotel**, serta mengeksplorasi pola pembatalannya. Dengan memvisualisasikan data ini, manajemen dapat memahami kecenderungan pemesanan mendadak dan membuat strategi yang tepat, misalnya dengan penawaran khusus atau kebijakan pembatalan yang lebih ketat untuk meminimalkan risiko kerugian.

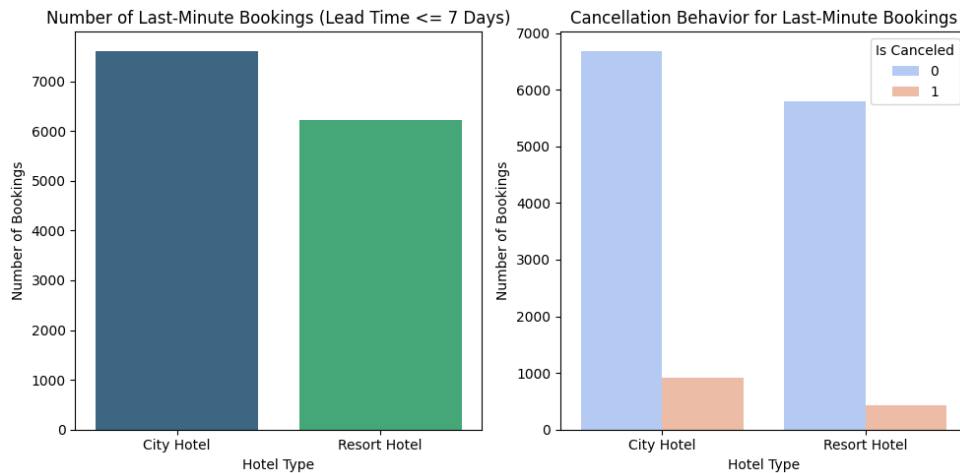
Kode Program SQL 11. Menghitung rasio pemesanan last-minute

```

1. # 1. Definisikan pemesanan Last-Minute sebagai lead_time <= 7 hari
2. last_minute_bookings = data[data['lead_time'] <= 7]
3.
4. # 2. Analisis berapa banyak pemesanan last-minute yang terjadi pada masing-
   masing hotel
5. last_minute_counts =
   last_minute_bookings['hotel'].value_counts().reset_index()
6. last_minute_counts.columns = ['hotel', 'count']
7.
8. # 3. Analisis perilaku pembatalan untuk pemesanan last-minute
9. canceled_last_minute = last_minute_bookings.groupby(['hotel',
   'is_canceled']).size().reset_index(name='count')
10.
11. # 4. Visualisasi distribusi pemesanan last-minute antara jenis hotel
12. plt.figure(figsize=(10, 5))
13.
14. plt.subplot(1, 2, 1)
15. sns.barplot(x='hotel', y='count', data=last_minute_counts,
   palette='viridis')
16. plt.title('Number of Last-Minute Bookings (Lead Time <= 7 Days)')
17. plt.ylabel('Number of Bookings')
18. plt.xlabel('Hotel Type')
19.
20. # 5. Visualisasi perilaku pembatalan untuk pemesanan last-minute
21. plt.subplot(1, 2, 2)
22. sns.barplot(x='hotel', y='count', hue='is_canceled',
   data=canceled_last_minute, palette='coolwarm')
23. plt.title('Cancellation Behavior for Last-Minute Bookings')
24. plt.ylabel('Number of Bookings')
25. plt.xlabel('Hotel Type')
26. plt.legend(title='Is Canceled', loc='upper right')
27.
28. # Menampilkan grafik
29. plt.tight_layout()
30. plt.show()

```

Visualisasi dari user behavior saat last minute sebelum check in



Grafik pertama menunjukkan jumlah pemesanan last-minute (lead time ≤ 7 hari) di dua jenis hotel: City Hotel dan Resort Hotel. City Hotel memiliki jumlah pemesanan last-minute yang lebih tinggi, sekitar 7.000 pemesanan, dibandingkan Resort Hotel yang mencapai sekitar 6.000 pemesanan. Ini mengindikasikan bahwa City Hotel lebih sering menarik tamu yang memesan dalam jangka waktu yang sangat dekat dengan tanggal check-in.

Grafik kedua memperlihatkan perilaku pembatalan untuk pemesanan last-minute di kedua jenis hotel. Pada City Hotel, meskipun ada lebih banyak pemesanan last-minute, tingkat pembatalan juga relatif tinggi. Terlihat sekitar 1.000 pemesanan dibatalkan, sedangkan 6.000 lainnya tidak dibatalkan. Di Resort Hotel, terdapat sekitar 800 pembatalan dari 6.000 pemesanan.

Kesimpulannya, baik City Hotel maupun Resort Hotel menunjukkan jumlah pemesanan last-minute yang signifikan, namun tingkat pembatalan di City Hotel sedikit lebih tinggi dibandingkan Resort Hotel.

4.2 A/B Testing

A/B testing adalah metode yang digunakan untuk membandingkan dua versi berbeda dari sesuatu, seperti web, iklan, produk, dan fitur. Dalam bidang data analysis atau data science A/B testing menjadi metode yang sangat penting untuk mengambil keputusan berdasarkan analisis data yang kuat.

Dalam penelitian ini, dilakukan pengujian A/B testing untuk menguji hipotesa terkait perbedaan rata-rata **Average Daily Rate (ADR)** antara City Hotel dan Resort Hotel. Hipotesa yang diuji adalah sebagai berikut:

- **H₀ (Hipotesa Nol):** Rata-rata ADR untuk City Hotel sama dengan Resort Hotel.
- **H₁ (Hipotesa Alternatif):** Rata-rata ADR untuk City Hotel berbeda dengan Resort Hotel.

Tujuan dari A/B testing ini adalah untuk menentukan apakah ada perbedaan yang signifikan secara statistik antara ADR kedua jenis hotel tersebut. A/B testing merupakan metode yang umum digunakan dalam analisis statistik untuk membandingkan dua kelompok (dalam hal ini City Hotel dan Resort Hotel) dan menguji apakah ada perbedaan yang nyata di antara mereka.

Langkah-langkah yang dilakukan dalam A/B testing ini meliputi:

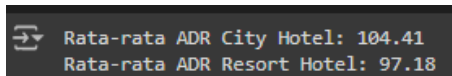
1) Menentukan nilai rata-rata adr berdasarkan masing-masing jenis hotel

Langkah awal dalam proses **A/B Testing** ini adalah menghitung nilai rata-rata **Average Daily Rate (ADR)** berdasarkan jenis hotel, yaitu **City Hotel** dan **Resort Hotel**. Tahap ini dilakukan untuk mendapatkan gambaran umum mengenai perbedaan tarif rata-rata harian pada kedua jenis hotel tersebut sebelum melanjutkan ke pengujian hipotesis yang lebih mendalam.

Kode Program SQL 11. Menghitung rata-rata adr

```
1. # Memisahkan ADR berdasarkan jenis hotel
2. city_hotel_adr = data[data['hotel'] == 'City Hotel']['adr']
3. resort_hotel_adr = data[data['hotel'] == 'Resort Hotel']['adr']
4.
5. # Rata-rata ADR masing-masing
6. print(f"Rata-rata ADR City Hotel: {city_hotel_adr.mean():.2f}")
7. print(f"Rata-rata ADR Resort Hotel: {resort_hotel_adr.mean():.2f}")
```

Output yang dihasilkan:



```
Rata-rata ADR City Hotel: 104.41
Rata-rata ADR Resort Hotel: 97.18
```

Berdasarkan perhitungan, rata-rata ADR untuk **City Hotel** adalah **104.41**, sedangkan untuk **Resort Hotel** adalah **97.18**. Perbedaan ini memberikan indikasi awal bahwa terdapat perbedaan harga antara kedua jenis hotel yang dapat digunakan sebagai dasar untuk analisis lebih lanjut mengenai tarif dan potensi strategi harga yang dapat diterapkan oleh masing-masing jenis hotel.

2) Melakukan pengujian hipotesa menggunakan metode uji T-Test

Pada bagian ini, dilakukan pengujian hipotesis menggunakan **uji T-Test independen** untuk mengukur apakah terdapat perbedaan signifikan pada rata-rata **Average Daily Rate (ADR)** antara **City Hotel** dan **Resort Hotel**. Uji T ini dilakukan dengan membandingkan dua kelompok data, yaitu ADR dari City Hotel dan ADR dari Resort Hotel.

Kode Program SQL 12. Pengujian Uji T-test

```
1. # Melakukan uji t-test independen
2. t_stat, p_value = stats.ttest_ind(city_hotel_adr, resort_hotel_adr)
3.
4. # Hasil uji t-test
5. print(f"T-Statistic: {t_stat:.4f}")
6. print(f"P-Value: {p_value:.4f}")
7.
8. # Menentukan hasil berdasarkan P-Value
9. alpha = 0.05
10. if p_value < alpha:
11.     print("Tolak H0: Ada perbedaan yang signifikan antara ADR City
        Hotel dan Resort Hotel.")
12. else:
13.     print("Terima H0: Tidak ada perbedaan yang signifikan antara ADR
        City Hotel dan Resort Hotel.")
```

Output yang dihasilkan:

```
T-Statistic: 29.4312
P-Value: 0.0000
Tolak H0: Ada perbedaan yang signifikan antara ADR City Hotel dan Resort Hotel.
```

Dalam kode, nilai T-statistik dan p-value dihitung menggunakan fungsi **T-Test** dari modul **scipy.stats**, yang mengasumsikan kedua kelompok data berdistribusi normal dan memiliki varians yang serupa. Hasil dari uji T menunjukkan **T-Statistic sebesar 29.4312** dan **p-value sebesar 0.0000**. Berdasarkan nilai p-value yang jauh lebih kecil dari tingkat signifikansi umum (misalnya, 0.05), keputusan yang diambil adalah **menolak H0**, yang berarti ada perbedaan signifikan pada rata-rata ADR antara City Hotel dan Resort Hotel.

Uji T-Test ini penting dalam konteks analisis bisnis, karena perbedaan signifikan dalam ADR mengindikasikan bahwa kedua tipe hotel memiliki struktur tarif harian yang berbeda, yang bisa jadi disebabkan oleh perbedaan target pasar, musim liburan, atau strategi pricing yang berbeda.

3) Melakukan pengujian hipotesa menggunakan metode Mann-Whitney U-Test

Pada bagian ini, digunakan **uji Mann-Whitney U-Test** untuk menguji hipotesis apakah terdapat perbedaan yang signifikan dalam distribusi **Average Daily Rate (ADR)** antara **City Hotel** dan **Resort Hotel**. Uji Mann-Whitney U digunakan sebagai alternatif dari uji

T-Test ketika asumsi normalitas pada data tidak terpenuhi. Hipotesis nol (H_0) dalam uji ini menyatakan bahwa distribusi ADR untuk City Hotel sama dengan distribusi ADR untuk Resort Hotel.

Kode Program SQL 13. Pengujian U-test

```
1. # 3. Mann-Whitney U Test (Alternatif dari T-Test)
2. # H0: Distribusi ADR untuk City Hotel = Distribusi ADR untuk Resort
   Hotel
3. u_stat, p_value = stats.mannwhitneyu(city_hotel_adr,
   resort_hotel_adr, alternative='two-sided')
4.
5. print(f"Mann-Whitney U Test Statistik: {u_stat}, P-Value: {p_value}")
6.
7. # Keputusan Hipotesis
8. alpha = 0.05
9. if p_value < alpha:
10.     print("Tolak H0: Ada perbedaan signifikan dalam distribusi ADR
       antara City Hotel dan Resort Hotel.")
11. else:
12.     print("Terima H0: Tidak ada perbedaan signifikan dalam distribusi
       ADR antara City Hotel dan Resort Hotel.")
13.
```

Output yang dihasilkan:

```
Mann-Whitney U Test Statistik: 1880005856.5, P-Value: 0.0
Tolak H0: Ada perbedaan signifikan dalam distribusi ADR antara City Hotel dan Resort Hotel.
```

Dalam kode, fungsi `mannwhitneyu` dari `scipy.stats` digunakan untuk menghitung nilai U-statistik dan p-value. Hasil dari uji ini menunjukkan U-Statistik sebesar 1,880,005,856.5 dan p-value sebesar 0.0. Karena p-value lebih kecil dari tingkat signifikansi (misalnya, 0.05), maka keputusan yang diambil adalah menolak H_0 , yang berarti terdapat perbedaan signifikan dalam distribusi ADR antara City Hotel dan Resort Hotel.

Penggunaan uji Mann-Whitney U ini memberikan wawasan yang lebih mendalam mengenai perbedaan distribusi ADR kedua jenis hotel, yang mungkin disebabkan oleh perbedaan dalam karakteristik pelanggan, strategi harga, atau faktor musiman yang memengaruhi ADR pada kedua jenis hotel tersebut.

BAB V

Kesimpulan & Saran

A. Kesimpulan utama

Studi ini berkontribusi besar pada peningkatan pemahaman tentang manajemen reservasi di industri perhotelan dengan menggunakan analitik data. Hasil kajian menunjukkan bahwa tingkat pembatalan reservasi yang tinggi, variasi lead time, kurangnya pengelolaan permintaan khusus, serta masalah dalam efisiensi pembayaran dan komunikasi selama proses reservasi adalah tantangan utama yang dihadapi oleh hotel. Dengan analisis ini, hotel dapat mengidentifikasi faktor-faktor yang menyebabkan ketidakpuasan pelanggan dan kerugian pendapatan. Temuan ini mendorong praktik bisnis yang lebih efektif, terutama dalam optimalisasi strategi manajemen pendapatan dan peningkatan layanan pelanggan melalui pengelolaan data yang lebih baik.

B. Saran untuk penelitian selanjutnya

Berikut ini adalah saran-saran untuk penelitian selanjutnya untuk mendapatkan analisis lebih dalam :

1. Mengeksplorasi pengaruh faktor musiman dan acara besar pada pola reservasi dan pembatalan.
2. Mengembangkan model prediksi tingkat pembatalan berdasarkan data historis untuk membantu hotel merancang strategi pencegahan.
3. Mengintegrasikan faktor-faktor eksternal seperti kebijakan harga kompetitor atau tren perjalanan lokal yang mungkin mempengaruhi keputusan pelanggan.
4. Menerapkan pendekatan machine learning yang lebih canggih untuk mempersonalisasi strategi manajemen permintaan khusus dan komunikasi dengan pelanggan.

BAB VI

Lampiran

A. Online Diagram

[**BPMN DIAGRAM**](#)

[**FLOW PROCESS**](#)

B. Python Code

[**PYTHON CODE - COLLAB**](#)

C. Recording

[**LINK RECORDING PRESENTATION**](#)