# 概要

## 用語定義

### リソースコード (ResourceCode) {#ResourceCode}

各種リソースの識別子として使用するコードを**リソースコード**と呼びます。 例えば、[施設](https://github.com/classmethod/metropolis-duck/blob/master/docs/src/apispec/content/resource/Facility.md)のリソースコードは、施設コードです。

### 予約リソース (ReservationResources) {#ReservationResources}

予約対象のサービスを提供するために必要となる要素を**予約リソース**と呼びます。 次のリソースが予約リソースです。

* [予約属性](https://github.com/classmethod/metropolis-duck/blob/master/docs/src/apispec/content/resource/ReservationAttribute.md)
* [提供場所](https://github.com/classmethod/metropolis-duck/blob/master/docs/src/apispec/content/resource/ProvisioningLocation.md)
* [スタッフ](https://github.com/classmethod/metropolis-duck/blob/master/docs/src/apispec/content/resource/Staff.md)
* [追加サービス](https://github.com/classmethod/metropolis-duck/blob/master/docs/src/apispec/content/resource/AdditionalService.md)

このうち[予約属性](https://github.com/classmethod/metropolis-duck/blob/master/docs/src/apispec/content/resource/ReservationAttribute.md)は**論理的な予約リソース**、 その他の予約リソースを**物理的な予約リソース**と呼びます。

### 確保 (Keep) {#Keep}

特定の時間に対して特定の[予約リソース](https://github.com/classmethod/metropolis-duck/blob/master/docs/src/apispec/content/abstract.md#ReservationResources)を、新しい[予約明細](https://github.com/classmethod/metropolis-duck/blob/master/docs/src/apispec/content/resource/ReservationDetail.md)に結びつけることを**確保**と呼びます。

予約サービスは、[スケジュール](https://github.com/classmethod/metropolis-duck/blob/master/docs/src/apispec/content/abstract.md#Schedule)や[キャパシティ](https://github.com/classmethod/metropolis-duck/blob/master/docs/src/apispec/content/abstract.md#Capacity)および現在までの予約状況に応じて、 [物理的な予約リソース](https://github.com/classmethod/metropolis-duck/blob/master/docs/src/apispec/content/abstract.md#ReservationResources)が不足しないように確保をします。物理的な予約リソースが不足する場合は確保が失敗します。

### アサイン (Assign)

履歴を元に、それぞれの予約に[物理的な予約リソース](https://github.com/classmethod/metropolis-duck/blob/master/docs/src/apispec/content/abstract.md#ReservationResources)を割り当てることを**アサイン**と呼びます。 しかし、予約サービス自体はアサインの確定までをサポートしません。

### 予約属性 (ReservationAttribute) {#ReservationAttribute}

サービスを実際に提供するには、最終的に[物理的な予約リソース](https://github.com/classmethod/metropolis-duck/blob/master/docs/src/apispec/content/abstract.md#ReservationResources)のアサインが必要です。 しかし予約の段階においては、複数の予約リソースから\_\_特定の特徴を持った\_\_物理的な予約リソースが確保できれば問題ありません。

予約サービスでは、予約の段階で物理的な予約リソースを確保するだけでなく、特定の特徴に対して[論理的な予約リソース](https://github.com/classmethod/metropolis-duck/blob/master/docs/src/apispec/content/abstract.md#ReservationResources)を確保できます。 この論理的な予約リソースを**予約属性**と呼びます。予約属性は、複数の物理的な予約リソースをグループ化する役割を持っています。

例えば、佐藤(男性) 鈴木(女性) 高橋(男性) という 3 人の[スタッフ](https://github.com/classmethod/metropolis-duck/blob/master/docs/src/apispec/content/resource/Staff.md)がいることを考えます。 1 人目の[予約者](https://github.com/classmethod/metropolis-duck/blob/master/docs/src/apispec/content/abstract.md#ReservationOwner)が男性スタッフによるサービスを希望した時、予約の段階で佐藤さんを確保してしまうと、 その後 2 人目の[予約者](https://github.com/classmethod/metropolis-duck/blob/master/docs/src/apispec/content/abstract.md#ReservationOwner)が佐藤さんを指名してサービスを希望しても、予約を受けられません。

そこで、1 人目の予約では佐藤さんを確保するのではなく、「男性スタッフ」という予約属性に対してリソースの確保を行います。 これにより最終的に、2 人目の予約に対しては佐藤さんをアサインし、さらに 1 人目の予約に対しては高橋さんのアサインが確定します。 ちなみに、この状態でさらに「佐藤さん指名」「高橋さん指名」「男性スタッフ希望」の予約は、いずれも受けられません。

予約属性は、この例に挙げた[スタッフ](https://github.com/classmethod/metropolis-duck/blob/master/docs/src/apispec/content/resource/Staff.md)だけでなく、他にも「特定の設備を備えた部屋」としての [提供場所](https://github.com/classmethod/metropolis-duck/blob/master/docs/src/apispec/content/resource/ProvisioningLocation.md) など、その他の物理的な予約リソースについても適用できます。

### 施設 (Facility)

予約を受け付けたサービスを提供する物理的な拠点を\*\*[施設](https://github.com/classmethod/metropolis-duck/blob/master/docs/src/apispec/content/resource/Facility.md)\*\*と呼びます。 営業時間等によって予約の可否が変わりますが、実際のサービス提供では施設を専有しません。 例えば、個々の店舗などが施設に相当します。

施設はその配下に、複数の[物理的な予約リソース](https://github.com/classmethod/metropolis-duck/blob/master/docs/src/apispec/content/abstract.md#ReservationResources)を持ちます。 つまり原則として、[物理的な予約リソース](https://github.com/classmethod/metropolis-duck/blob/master/docs/src/apispec/content/abstract.md#ReservationResources)はそれぞれ 1 つの施設に関連付きます。

ただし、[スタッフ](https://github.com/classmethod/metropolis-duck/blob/master/docs/src/apispec/content/abstract.md#Staff)は施設との関連に有効期間を持ち、この期間が重ならない限り複数の施設に所属できます。

### 提供場所 (ProvisioningLocation)

予約を受け付けたサービスを提供する際に専有する場所を\*\*[提供場所](https://github.com/classmethod/metropolis-duck/blob/master/docs/src/apispec/content/resource/ProvisioningLocation.md)\*\*と呼びます。 例えば、個々の施術台や個室などが提供場所に相当します。

### スタッフ (Staff) {#Staff}

接客や在庫管理、事務など、[施設](https://github.com/classmethod/metropolis-duck/blob/master/docs/src/apispec/content/resource/Facility.md)でのサービス運営・管理に関わる業務を行う人を\*\*[スタッフ](https://github.com/classmethod/metropolis-duck/blob/master/docs/src/apispec/content/resource/Staff.md)\*\*と呼びます。

### 施設管理者 (FacilitiesManager) {#FacilitiesManager}

スタッフの中でも、売上や労務、施設管理など、[施設](https://github.com/classmethod/metropolis-duck/blob/master/docs/src/apispec/content/resource/Facility.md)でのサービス運営・管理に責任を持つ人のことを**施設管理者**と呼びます。 施設管理者は、[スタッフ](https://github.com/classmethod/metropolis-duck/blob/master/docs/src/apispec/content/abstract.md#Staff)の一種です。

### 予約者 (ReservationOwner) {#ReservationOwner}

[予約リソース](https://github.com/classmethod/metropolis-duck/blob/master/docs/src/apispec/content/abstract.md#ReservationResources)を確保するための予約申し込みを行う会員およびビジターを**予約者**と呼びます。 予約者は会員サービスを使用して予約者情報を登録します。 会員サービスでは、既に会員登録している「会員」と、電話予約や一時ユーザとしてネット予約を行う「非会員」とが混在するようになりますが、 会員タイプによって会員と非会員を判別できるようにします。 非会員の予約者が会員登録する際には、新たに会員サービスで会員リソースを作成し、会員統合で既存の予約者情報と統合します。

### 利用者 (User) {#User}

予約したサービスを利用する会員およびビジターのことを**利用者**と呼びます。 利用者の[利用実績](https://github.com/classmethod/metropolis-duck/blob/master/docs/src/apispec/content/abstract.md#UseResult)を記録する場合、会員サービスを使用して利用者情報を登録します。 既に会員登録を行っている「会員」と、会員登録をしていない「非会員」とが混在するようになりますが、会員タイプによって会員と非会員を判別できるようにします。 非会員の利用者が会員登録する際には、新たに会員サービスで会員リソースを作成し、会員統合で既存の利用者情報と統合します。

### 予約申込日時 (ReservationApplicationDatetime) {#ReservationApplicationDatetime}

[予約者](https://github.com/classmethod/metropolis-duck/blob/master/docs/src/apispec/content/abstract.md#ReservationOwner)が[予約リソース](https://github.com/classmethod/metropolis-duck/blob/master/docs/src/apispec/content/abstract.md#ReservationResources)を確保するために予約申し込みを行った日時のことを**予約申込日時**と呼びます。 具体的には[予約明細](https://github.com/classmethod/metropolis-duck/blob/master/docs/src/apispec/content/resource/ReservationDetail.md)作成をリクエストした日時を指し、予約受付判定や[価格の決定](https://github.com/classmethod/metropolis-duck/blob/master/docs/src/apispec/content/calculation-process/determine-price.md)に使用します。

### 予約対象日時 (ReservationsTargetDatetime) {#ReservationsTargetDatetime}

[予約者](https://github.com/classmethod/metropolis-duck/blob/master/docs/src/apispec/content/abstract.md#ReservationOwner)が予約する対象日時を**予約対象日時**と呼びます。 具体的には[予約](https://github.com/classmethod/metropolis-duck/blob/master/docs/src/apispec/content/resource/Reservation.md)の予約対象開始タイムスタンプ .reservation\_target\_start\_timestamp を指し、[予約者](https://github.com/classmethod/metropolis-duck/blob/master/docs/src/apispec/content/abstract.md#ReservationOwner)がリクエストした予約対象サービスの予約受付判定や[価格の決定](https://github.com/classmethod/metropolis-duck/blob/master/docs/src/apispec/content/calculation-process/determine-price.md)に使用します。

### 予約受付日時 (ReservationsAcceptedDatetime) {#ReservationsAcceptedDatetime}

[予約者](https://github.com/classmethod/metropolis-duck/blob/master/docs/src/apispec/content/abstract.md#ReservationOwner)がリクエストした[予約](https://github.com/classmethod/metropolis-duck/blob/master/docs/src/apispec/content/resource/Reservation.md)を確定し、予約手続きが完了した日時のことを**予約受付日時**と呼びます。 [予約リソース](https://github.com/classmethod/metropolis-duck/blob/master/docs/src/apispec/content/abstract.md#ReservationResources)の確保や決済など予約に必要な手続きを終え、予約を確定する際に [UpdateReservationAcceptedTimestamp](https://github.com/classmethod/metropolis-duck/blob/master/docs/src/apispec/content/api/UpdateReservationAcceptedTimestamp.md) を実行することで予約受付日時を記録します。

### 基準時刻 (ReferenceTime) {#ReferenceTime}

1 日を区切る時刻のことを**基準時刻**と呼びます。 仮に基準時刻を 10 時にした場合、7 月 7 日 10:00:00:000 〜 翌日の 7 月 8 日 09:59:59:999 までを 1 日とします。 この時、7 月 8 日 09:00 に売上が発生した場合、売上計上日は 7 月 7 日となります。

基準時刻は、[施設](https://github.com/classmethod/metropolis-duck/blob/master/docs/src/apispec/content/resource/Facility.md)および[予約リソース](https://github.com/classmethod/metropolis-duck/blob/master/docs/src/apispec/content/abstract.md#ReservationResources)が持つプロパティです。

### 明細リソース (DetailResource) {#DetailResource}

提供するサービス単位で金額や数量を細かく定義しているものを**明細リソース**と呼びます。 次のリソースが明細リソースです。

* [予約明細](https://github.com/classmethod/metropolis-duck/blob/master/docs/src/apispec/content/resource/ReservationDetail.md)
* [追加リソース明細](https://github.com/classmethod/metropolis-duck/blob/master/docs/src/apispec/content/resource/AdditionalResourceDetail.md)

### 利用料金/合計利用料金 (UsingPrice) {#UsingPrice}

サービスの利用にかかる 1 予約または利用あたりの金額を**利用料金**と呼び、予約数や利用数により最終的にかかる金額を**合計利用料金**と呼びます。 利用料金はランク料や指名料などの追加料金を含みません。合計利用料金は[明細リソース](https://github.com/classmethod/metropolis-duck/blob/master/docs/src/apispec/content/abstract.md#DetailResource)から次のように算出します。

* [予約条件](https://github.com/classmethod/metropolis-duck/blob/master/docs/src/apispec/content/resource/ReservationCondition.md) が関連付いている場合
  1. 予約条件リソース指定情報 ReservationConditionResourceInfo に関連付いている[価格](https://github.com/classmethod/metropolis-duck/blob/master/docs/src/apispec/content/resource/PriceList.md#Price)が**利用料金**です。
  2. 利用料金 × 予約数または利用数 の計算結果が**合計利用料金**です。
* 例: 予約条件に関連付く価格が 8000 円、予約数が 2 の場合
* 利用料金は 8000 円、合計利用料金は 16000 円です。
* 直接リソースを指定している場合
  1. 直接リソース指定情報の予約属性情報 DirectReservationAttribute または予約リソース情報 DirectResource に関連付いている[価格](https://github.com/classmethod/metropolis-duck/blob/master/docs/src/apispec/content/resource/PriceList.md#Price)と数量 .quantity を使用します。
  2. 価格 × 数量 の計算結果が**利用料金**です。
  3. 利用料金 × 予約数または利用数 の計算結果が**合計利用料金**です。
* 例: 追加サービスに関連付く価格が 30 分あたり 1500 円、数量が 2 (リソースを 1 時間利用する)、予約数が 2 の場合
* 利用料金は 3000 円、合計利用料金は 6000 円です。

### 追加料金/合計追加料金 (AdditionalPrice) {#AdditionalPrice}

ランク料や指名料などある特定の[物理的な予約リソース](https://github.com/classmethod/metropolis-duck/blob/master/docs/src/apispec/content/abstract.md#ReservationResources)を指定した時にかかる 1 リソースあたりの金額を**追加料金**と呼び、 予約数や利用数により最終的にかかる金額を**合計追加料金**と呼びます。 合計追加料金は[明細リソース](https://github.com/classmethod/metropolis-duck/blob/master/docs/src/apispec/content/abstract.md#DetailResource)から次のように算出します。

* [予約条件](https://github.com/classmethod/metropolis-duck/blob/master/docs/src/apispec/content/resource/ReservationCondition.md) が関連付いている場合
  1. 予約条件リソース指定情報の予約リソース情報 ReservationConditionResource に関連付いている[価格](https://github.com/classmethod/metropolis-duck/blob/master/docs/src/apispec/content/resource/PriceList.md#Price)が**追加料金**です。
  2. 追加料金 × リソースの確保数 の計算結果が**合計追加料金**です。
* 例: 追加サービスに関連付いている価格が 500 円、リソースの確保数が 2 の場合
* 追加料金は 500 円、合計追加料金は 1000 円です。
* 直接リソースを指定している場合
  1. 直接リソース指定情報の予約リソース情報 DirectResource に関連付いている合計追加料金の[価格](https://github.com/classmethod/metropolis-duck/blob/master/docs/src/apispec/content/resource/PriceList.md#Price)が**追加料金**です。
  2. 追加料金 × 予約数または利用数 の計算結果が**合計追加料金**です。
* 例: 追加サービスに関連付いている価格が 100 円、リソースの確保数が 4 の場合
* 追加料金は 100 円、合計追加料金は 400 円です。

## 税金

**税金**では、租税として納める金額そのものや、金額を算出するための税率といった情報を持ちます。 「消費税」の場合、「税率」を入力して税金を作成します。 「入湯税」のように税額が固定の場合は、「税額」に金額を入力して税金を作成します。

税金は次のリソースと関連付きます。

* [予約属性](https://github.com/classmethod/metropolis-duck/blob/master/docs/src/apispec/content/resource/ReservationAttribute.md)
* [提供場所](https://github.com/classmethod/metropolis-duck/blob/master/docs/src/apispec/content/resource/ProvisioningLocation.md)
* [スタッフ](https://github.com/classmethod/metropolis-duck/blob/master/docs/src/apispec/content/resource/Staff.md)
* [追加サービス](https://github.com/classmethod/metropolis-duck/blob/master/docs/src/apispec/content/resource/AdditionalService.md)
* [予約条件](https://github.com/classmethod/metropolis-duck/blob/master/docs/src/apispec/content/resource/ReservationCondition.md)

1 つの税金は、税金を扱う複数のリソースと関連付きます。 1 つの税金を扱うリソースは、複数の税金と関連付きます。 「消費税」の場合、税金「消費税」を作成し、消費税を扱うリソースと関連付けます。 「お酒」のように「酒税」と「消費税」といった複数の税金を持つ場合は、それぞれを税金として作成し、「お酒」に関連付けます。

美容室であれば、提供サービスに対する消費税加算のため、すべての予約条件に税金「消費税」を関連付けます。 さらにスタッフ指名料に対する消費税を加算するため、[スタッフ](https://github.com/classmethod/metropolis-duck/blob/master/docs/src/apispec/content/resource/Staff.md)「佐藤」「鈴木」「高橋」「田中」「渡辺」「山本」「中村」「小林」に 税金「消費税」を関連付けます。

税率変更時は、終了予定となる税金の適用終了日を更新し、税率と適用期間が異なる新しい税金を作成します。

### 案分係数 (DistributionCoefficient) {#DistributionCoefficient}

**案分係数**とは、予約条件で設定した価格を基に[予約リソース](https://github.com/classmethod/metropolis-duck/blob/master/docs/src/apispec/content/abstract.md#ReservationResources)毎の売上を算出するための係数です。 案分係数は、[予約属性](https://github.com/classmethod/metropolis-duck/blob/master/docs/src/apispec/content/resource/ReservationAttribute.md)やリソースグループ単位で算出します。予約条件の価格に案分係数を乗算した結果が、[予約リソース](https://github.com/classmethod/metropolis-duck/blob/master/docs/src/apispec/content/abstract.md#ReservationResources)の売上です。 案分係数の算出方法は「[案分係数の算出](https://github.com/classmethod/metropolis-duck/blob/master/docs/src/apispec/content/calculation-process/calculate-daily-sales-amount.md#CalcDistributionCoefficient)」を参照してください。

### キャパシティ (Capacity) {#Capacity}

同一の時間帯 (稼働開始時刻と稼働終了時刻の組み合わせ) に受付可能な予約数を**キャパシティ**と呼びます。 キャパシティは、[スケジュール](https://github.com/classmethod/metropolis-duck/blob/master/docs/src/apispec/content/resource/Schedule.md)や[予約可能枠](https://github.com/classmethod/metropolis-duck/blob/master/docs/src/apispec/content/resource/ReservableFrame.md)に対して付与するプロパティです。

### スケジュール (Schedule) {#Schedule}

[物理的な予約リソース](https://github.com/classmethod/metropolis-duck/blob/master/docs/src/apispec/content/abstract.md#ReservationResources)および[施設](https://github.com/classmethod/metropolis-duck/blob/master/docs/src/apispec/content/resource/Facility.md)が持っている稼働可能な時間のことを \*\*[スケジュール](https://github.com/classmethod/metropolis-duck/blob/master/docs/src/apispec/content/resource/Schedule.md)\*\*と呼びます。

[物理的な予約リソース](https://github.com/classmethod/metropolis-duck/blob/master/docs/src/apispec/content/abstract.md#ReservationResources)および[施設](https://github.com/classmethod/metropolis-duck/blob/master/docs/src/apispec/content/resource/Facility.md)は、稼働時間が異なる複数のスケジュールに関連付きます。 1 つのスケジュールは、1 つの[物理的な予約リソース](https://github.com/classmethod/metropolis-duck/blob/master/docs/src/apispec/content/abstract.md#ReservationResources)または[施設](https://github.com/classmethod/metropolis-duck/blob/master/docs/src/apispec/content/resource/Facility.md)に関連付きます。

1 つのスケジュールには[キャパシティ](https://github.com/classmethod/metropolis-duck/blob/master/docs/src/apispec/content/abstract.md#Capacity)を設定します。 [予約条件](https://github.com/classmethod/metropolis-duck/blob/master/docs/src/apispec/content/resource/ReservationCondition.md)で設定した[物理的な予約リソース](https://github.com/classmethod/metropolis-duck/blob/master/docs/src/apispec/content/abstract.md#ReservationResources)に対して、スケジュールを作成する必要があります。

例えば、美容室であれば「[施設](https://github.com/classmethod/metropolis-duck/blob/master/docs/src/apispec/content/resource/Facility.md)」「[提供場所](https://github.com/classmethod/metropolis-duck/blob/master/docs/src/apispec/content/resource/ProvisioningLocation.md)」「[スタッフ](https://github.com/classmethod/metropolis-duck/blob/master/docs/src/apispec/content/resource/Staff.md)」それぞれに対するスケジュールを作成します。 次の例では、2018 年 7 月 1 日のスケジュールを作成します。

### スケジュールテンプレート

**スケジュールテンプレート** は[スケジュール](https://github.com/classmethod/metropolis-duck/blob/master/docs/src/apispec/content/resource/Schedule.md)を繰り返し登録する場合に利用するリソースです。 本リソースを用いることで指定期間内に同じ[スケジュール](https://github.com/classmethod/metropolis-duck/blob/master/docs/src/apispec/content/resource/Schedule.md)を繰り返し作成できます。

### スケジュール確定 (ScheduleConfirmation)

[物理的な予約リソース](https://github.com/classmethod/metropolis-duck/blob/master/docs/src/apispec/content/abstract.md#ReservationResources)に関連付く[スケジュール](https://github.com/classmethod/metropolis-duck/blob/master/docs/src/apispec/content/resource/Schedule.md)を[予約可能枠](https://github.com/classmethod/metropolis-duck/blob/master/docs/src/apispec/content/resource/ReservableFrame.md)の計算対象とし、 外部公開することを**スケジュール確定**と呼びます。 [スケジュール確定](https://github.com/classmethod/metropolis-duck/blob/master/docs/src/apispec/content/resource/ScheduleConfirmation.md)リソースでは、スケジュール確定対象の[予約リソース](https://github.com/classmethod/metropolis-duck/blob/master/docs/src/apispec/content/abstract.md#ReservationResources)と確定する期間を指定します。 スケジュール確定により、指定した期間内のスケジュールを予約可能枠の計算対象にし、[予約者](https://github.com/classmethod/metropolis-duck/blob/master/docs/src/apispec/content/abstract.md#ReservationOwner)が予約可能な状態にします。スケジュール確定後もスケジュールの更新は可能です。

予約可能枠の計算は、計算時点のスケジュールを参照します。スケジュール確定後のスケジュール変更により、[物理的な予約リソース](https://github.com/classmethod/metropolis-duck/blob/master/docs/src/apispec/content/abstract.md#ReservationResources)が不足して 予約可能枠がマイナスになる可能性がありますが、予約サービスでは不足に対する制御はせずに不足している状態をそのまま返却します。不足が出る場合、クライアント側で調整する必要があります。

1 つの[スケジュール確定](https://github.com/classmethod/metropolis-duck/blob/master/docs/src/apispec/content/resource/ScheduleConfirmation.md)は 1 つの[予約リソース](https://github.com/classmethod/metropolis-duck/blob/master/docs/src/apispec/content/abstract.md#ReservationResources)に関連付きます。 1 つの予約リソースは、複数のスケジュール確定が関連付きます。

#### スケジュール確定のイメージ

以下の[スケジュール](https://github.com/classmethod/metropolis-duck/blob/master/docs/src/apispec/content/resource/Schedule.md)を想定します。

| **スケジュール** | **リソース** | **稼働開始日時** | **稼働終了日時** |
| --- | --- | --- | --- |
| スケジュール A1 | 施設 A | 2019/04/01 10:00:00 | 2019/04/01 19:00:00 |
| スケジュール A2 | 施設 A | 2019/04/02 10:00:00 | 2019/04/02 19:00:00 |
| スケジュール A3 | 施設 A | 2019/06/01 10:00:00 | 2019/06/01 19:00:00 |
| スケジュール B1 | 施設 B | 2019/10/01 10:00:00 | 2019/10/01 19:00:00 |

[スケジュール確定](https://github.com/classmethod/metropolis-duck/blob/master/docs/src/apispec/content/resource/ScheduleConfirmation.md)を以下に示します。

| **スケジュール確定** | **リソース** | **確定期間** | **説明** |
| --- | --- | --- | --- |
| スケジュール確定 A1 | 施設 A | 2019/04/01 00:00:00 〜 2019/05/01 00:00:00 | 施設 A の 4 月分のスケジュール確定 |
| スケジュール確定 A2 | 施設 A | 2019/05/01 00:00:00 〜 2019/06/01 00:00:00 | 施設 A の 5 月分のスケジュール確定 |
| スケジュール確定 B1 | 施設 B | 2019/03/01 00:00:00 〜 2019/06/01 00:00:00 | 施設 B の 3 月から 3 か月分のスケジュール確定 |

施設 A は スケジュール確定 A1 と スケジュール確定 A2 を持つため、確定期間は 2019/04/01〜2019/06/01 になります。

確定期間内にある スケジュール A1 と スケジュール A2 がスケジュール確定の対象になります。 スケジュール A3 は確定期間内ではないため、スケジュール確定の対象外です。

#### 予約可能枠のイメージ

| **日時** | **3 月 1 日** | **3 月 2 日** | **3 月 3 日** |
| --- | --- | --- | --- |
| 10 : 00 | 予約可能枠 A (キャパシティ: 2) | 予約可能枠 H (キャパシティ: 2) | 予約可能枠 O (キャパシティ: 2) |
| 11 : 00 | 予約可能枠 B (キャパシティ: 3) | 予約可能枠 I (キャパシティ: 3) | 予約可能枠 P (キャパシティ: 3) |
| 12 : 00 | 予約可能枠 C (キャパシティ: 3) | 予約可能枠 J (キャパシティ: 3) | 予約可能枠 Q (キャパシティ: 3) |
| ... | ... | ... | ... |

予約サービスでは、スケジュール、予約条件、予約から予約可能枠を導出します。 スケジュールの確保や開放を行うことで予約可能枠を更新します。

予約可能枠は予約サービス側で計算せず、リソースやキャパシティを直接指定して作成も可能です。 予約可能枠を作成した場合、予約サービスでスケジュールの確保や開放による予約可能枠の更新を行いません。 予約可能枠の更新 API を使用して、クライアント側でキャパシティの管理を行なってください。

### 利用実績/利用実績リソース {#UseResult}

利用者の施設やサービス利用の記録を**利用実績**と呼び、利用実績を管理するリソース群を**利用実績リソース**と呼びます。 次のリソースが利用実績リソースです。

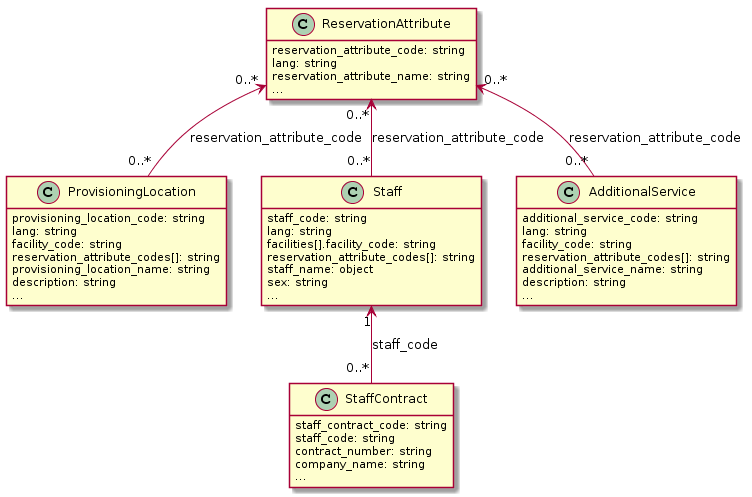
* [チェックイン](https://github.com/classmethod/metropolis-duck/blob/master/docs/src/apispec/content/resource/CheckIn.md)
* [追加リソース明細](https://github.com/classmethod/metropolis-duck/blob/master/docs/src/apispec/content/resource/AdditionalResourceDetail.md)
* 利用登録
* [チェックアウト](https://github.com/classmethod/metropolis-duck/blob/master/docs/src/apispec/content/resource/Checkout.md)
* [締め情報](https://github.com/classmethod/metropolis-duck/blob/master/docs/src/apispec/content/resource/ClosingInfo.md)

## 概念モデル

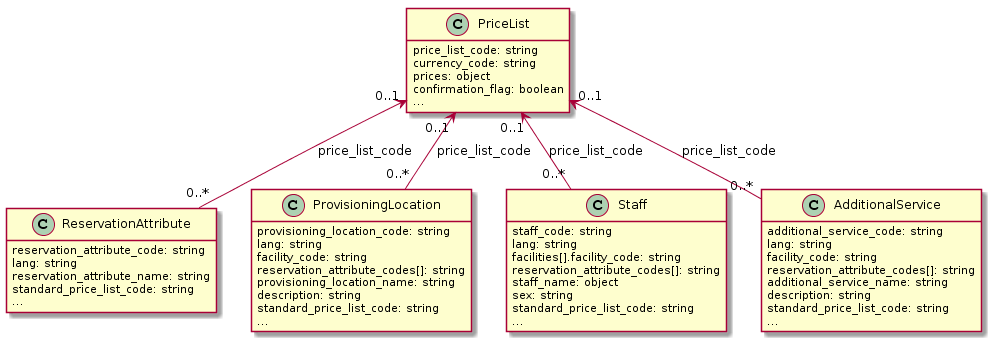
静的な概念図は次の通りです。

### 予約リソースの概念図

「男性スタッフ指名」のようなリソースを直接指定しない予約を受け付けるために、物理的な予約リソースを予約属性と関連付ける必要があります。

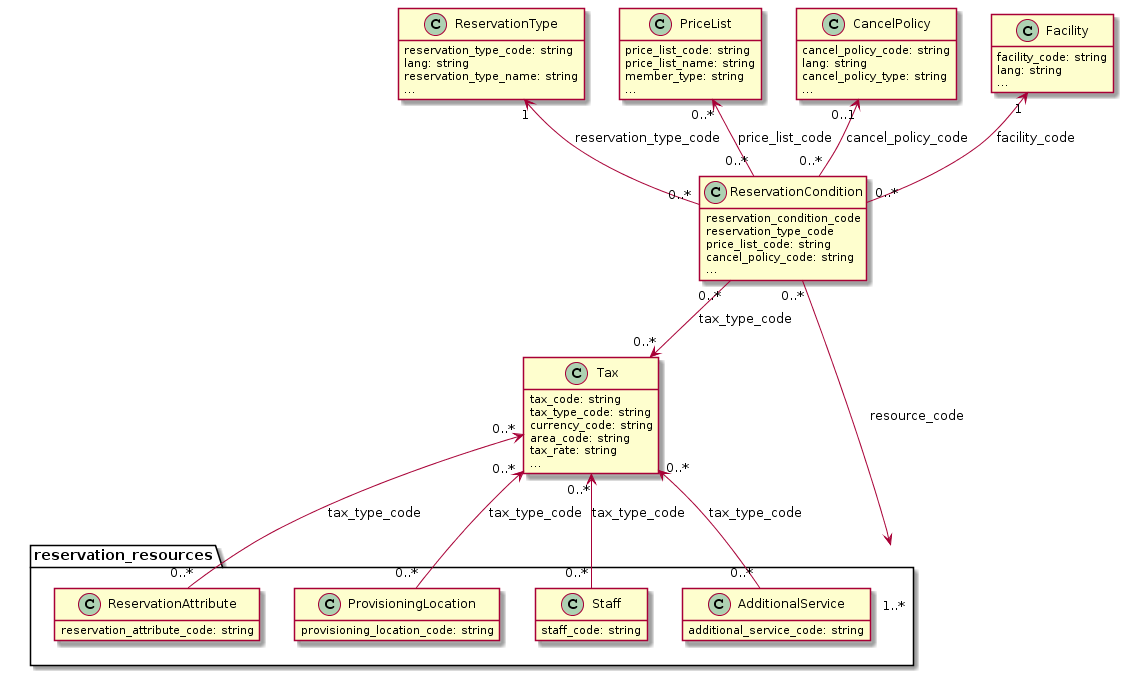


予約リソースは、価格表と関連付きます。 予約リソースで関連付けた価格はリソースの標準価格を表し、予約条件を介さず直接リソースを指定して予約する場合の価格として使用します。 1 つの予約リソースに価格表が関連付かない場合もあります。1 つの価格表は、複数の予約リソースに関連付く場合があります。



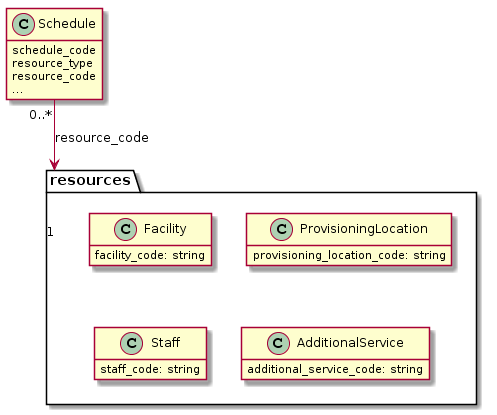
### 予約条件の概念図

予約条件では、予約に必要なリソースを確保するため、予約属性や施設などの予約リソースを指定します。 1 つの予約条件で、1 つ以上の予約リソースに関連付きます。 予約リソースと予約条件は、税金と関連付きます。関連付けた税金は、予約金額や請求金額を算出する時に使用します。



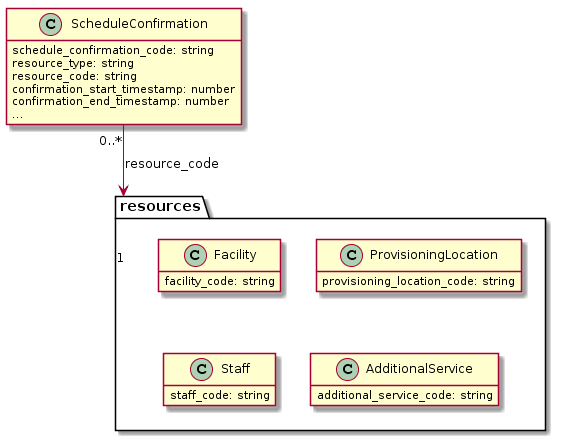
### スケジュールの概念図

スケジュールは、予約可能枠を生成するため、予約条件で指定した物理的な予約リソースと関連付ける必要があります。



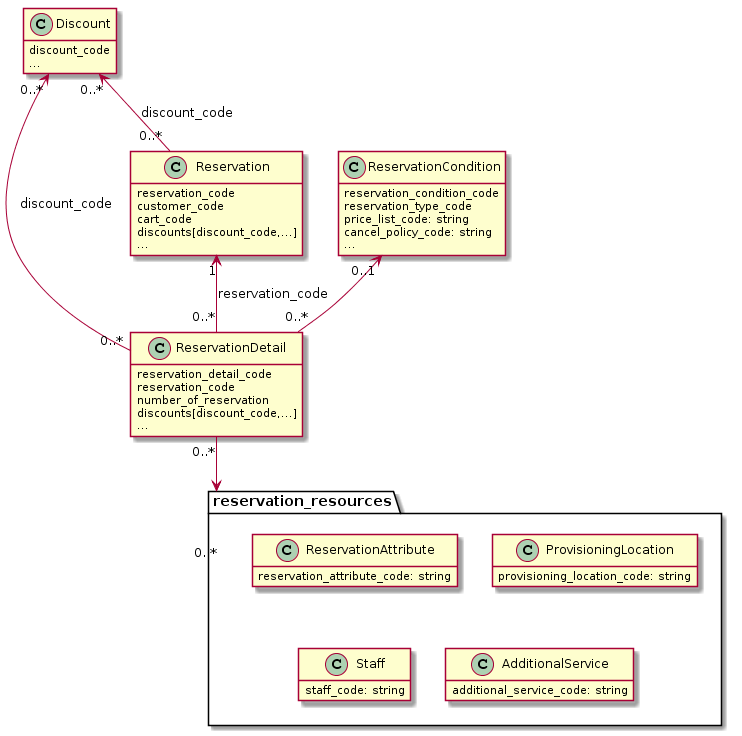
### スケジュール確定の概念図

スケジュール確定は、指定したリソースのスケジュールを確定する期間を定義します。 本リソースは予約可能枠を計算する際に利用します。



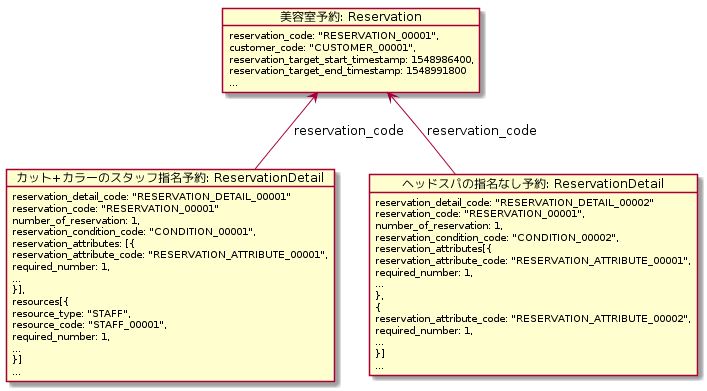
### 予約の概念図

予約には、どのリソースをどれだけ確保するといった予約明細を複数関連付けます。 スケジュール確保 API を実行することでリソースを確保します。 1 件の予約に対して 10% 割引するといった予約全体に割引を適用するには、予約と割引を関連付けます。 特定のリソースに対して割引を適用するには、予約明細と割引を関連付けます。



### 予約明細の概念図

予約明細は、1 つの予約に対して複数関連付けられます。 例えば、美容室で「カット + カラーのスタッフ指名予約」という予約条件や「ヘッドスパ」といった予約条件など、複数の予約明細を作成し関連付けられます。 予約明細は予約条件毎の作成を基本としますが、予約条件で定義していない予約明細も作成できます。



### 予約開始の概念図

予約開始は、対象日の予約受付期間を定義します。 予約サービス利用者が予約を行う際に、予約しようとしている対象日が受付可能であるかを判断します。 そのため、予約種類と関連があります。

