

# hacomono 予約システム 必須条件・制約まとめ

## 概要

hacomono APIには2種類の予約方式があります：

予約タイプ	エンドポイント	特徴
固定枠予約（レッスン）	POST /reservation/reservations/reserve	事前に作成したレッスン枠に対して予約
自由枠予約 (choice)	POST /reservation/reservations/choice/reserve	営業時間内で自由に時間を選んで予約

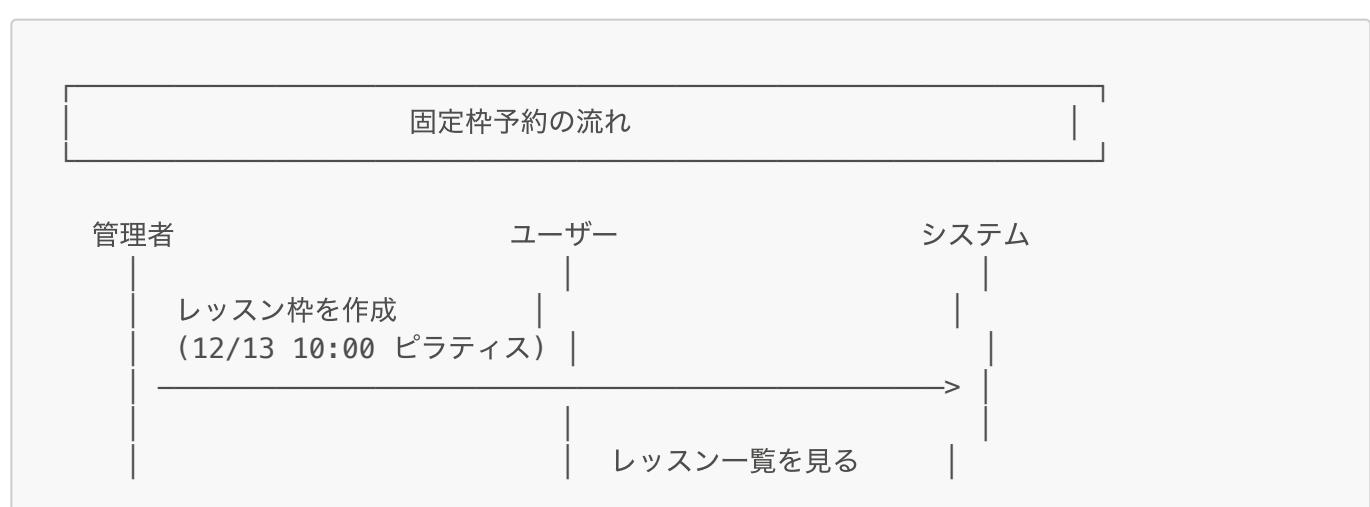
## 固定枠予約 vs 自由枠予約（詳細比較）

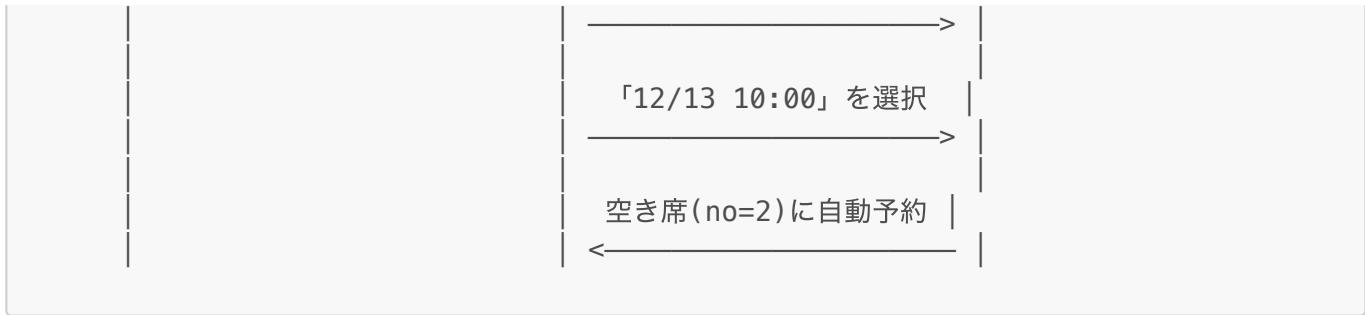
基本的な違い

項目	固定枠予約（レッスン予約）	自由枠予約（choice予約）
API	POST /reservation/reservations/reserve	POST /reservation/reservations/choice/reserve
特徴	事前に作成したレッスン枠に予約	営業時間内で自由に時間を選んで予約
時間選択	レッスン枠で決まっている	ユーザーが開始時間を指定
席番号 (no)	必要（自動で空き席を選択）	不要
用途	グループレッスン、定時クラス	パーソナルトレーニング、個別予約

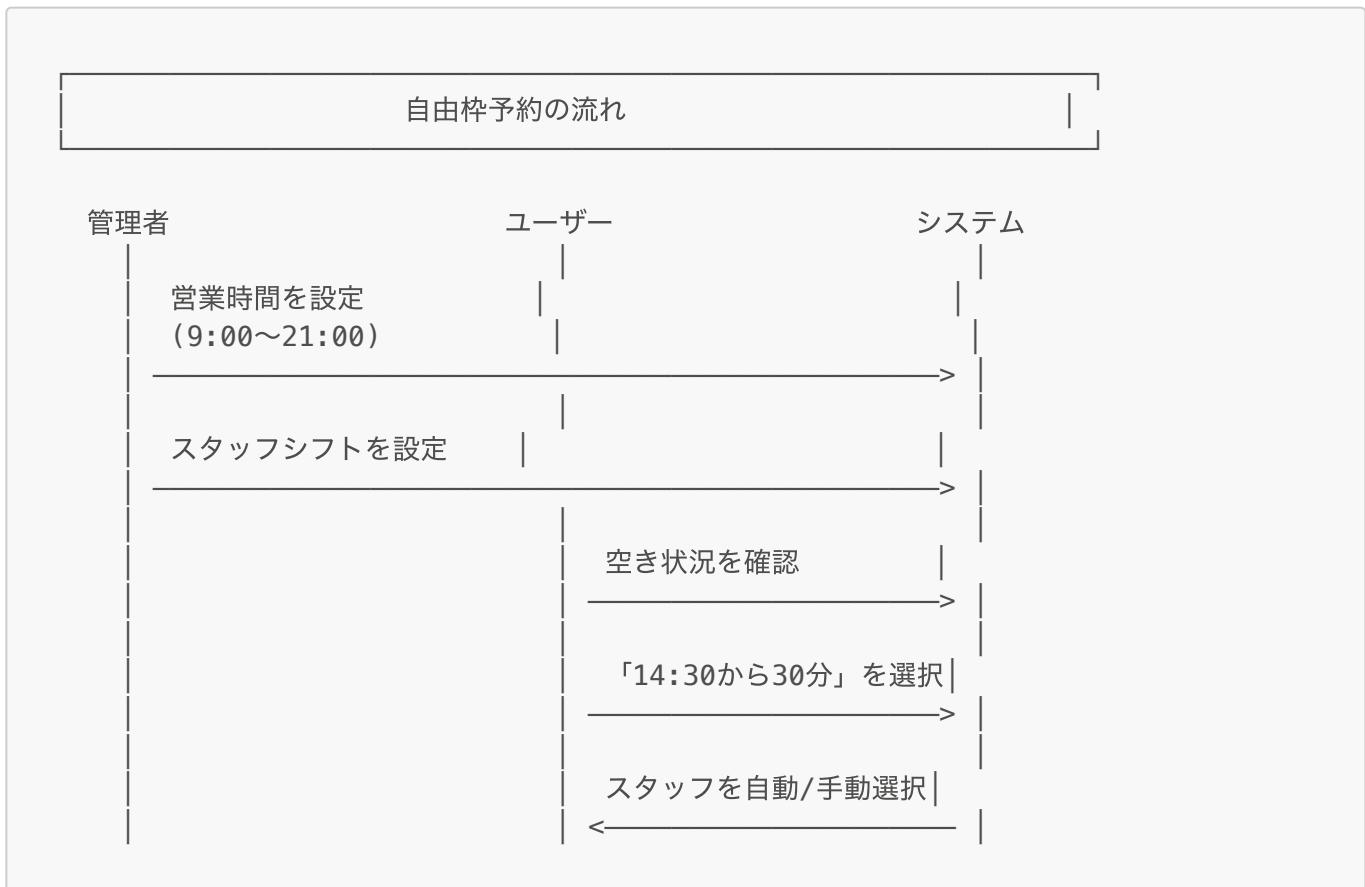
予約の流れ

### 固定枠予約の流れ





## 自由枠予約の流れ



## 必要なAPIパラメータ

### 固定枠予約

```
{
  "member_id": 49,
  "studio_lesson_id": 248,      // ← レッスン枠ID（必須）
  "no": "2",                   // ← 席番号（自動選択）
  "member_ticket_id": 43
}
```

### 自由枠予約

```
{
  "member_id": 34,
  "studio_room_id": 3,           // ← 部屋ID (必須)
  "program_id": 2,              // ← プログラムID (必須)
  "start_at": "2025-12-13 14:30:00.000", // ← 自分で指定
  "instructor_ids": [1],        // ← スタッフID (必須)
  "ticket_id": 5
}
```

## 管理画面での設定の違い

設定項目	固定枠予約	自由枠予約
スタジオルーム作成	<input checked="" type="checkbox"/> 予約タイプ: 「固定枠」	<input checked="" type="checkbox"/> 予約タイプ: 「自由枠」
レッスン枠作成	<input checked="" type="checkbox"/> 必須	<input checked="" type="checkbox"/> 不要
スペース設定 (席番号)	<input checked="" type="checkbox"/> 必須 (noフィールド)	<input checked="" type="checkbox"/> 不要
月次シフト作成	△ 任意	<input checked="" type="checkbox"/> 必須
営業時間設定	△ 任意	<input checked="" type="checkbox"/> 必須
スタッフシフト設定	△ 任意	<input checked="" type="checkbox"/> 必須
ルームサービス設定	<input checked="" type="checkbox"/> 不要	<input checked="" type="checkbox"/> 必須 (期間設定)

## 現在の設定 (Test Studio)

部屋名	タイプ	studio_room_id	備考
Pilates Room	固定枠予約	5	レッスン枠から予約
Test Room	自由枠予約	3 or 4	時間を自由に選択

## どちらを使うべき？

ユースケース	推奨
グループレッスン (ヨガ、ピラティスなど)	<b>固定枠予約</b>
定員のあるクラス	<b>固定枠予約</b>
パーソナルトレーニング	<b>自由枠予約</b>
施術・エステなど個別予約	<b>自由枠予約</b>
スタッフが時間を柔軟に対応	<b>自由枠予約</b>

## ゲスト予約の流れ (重要)

### チケットとは？

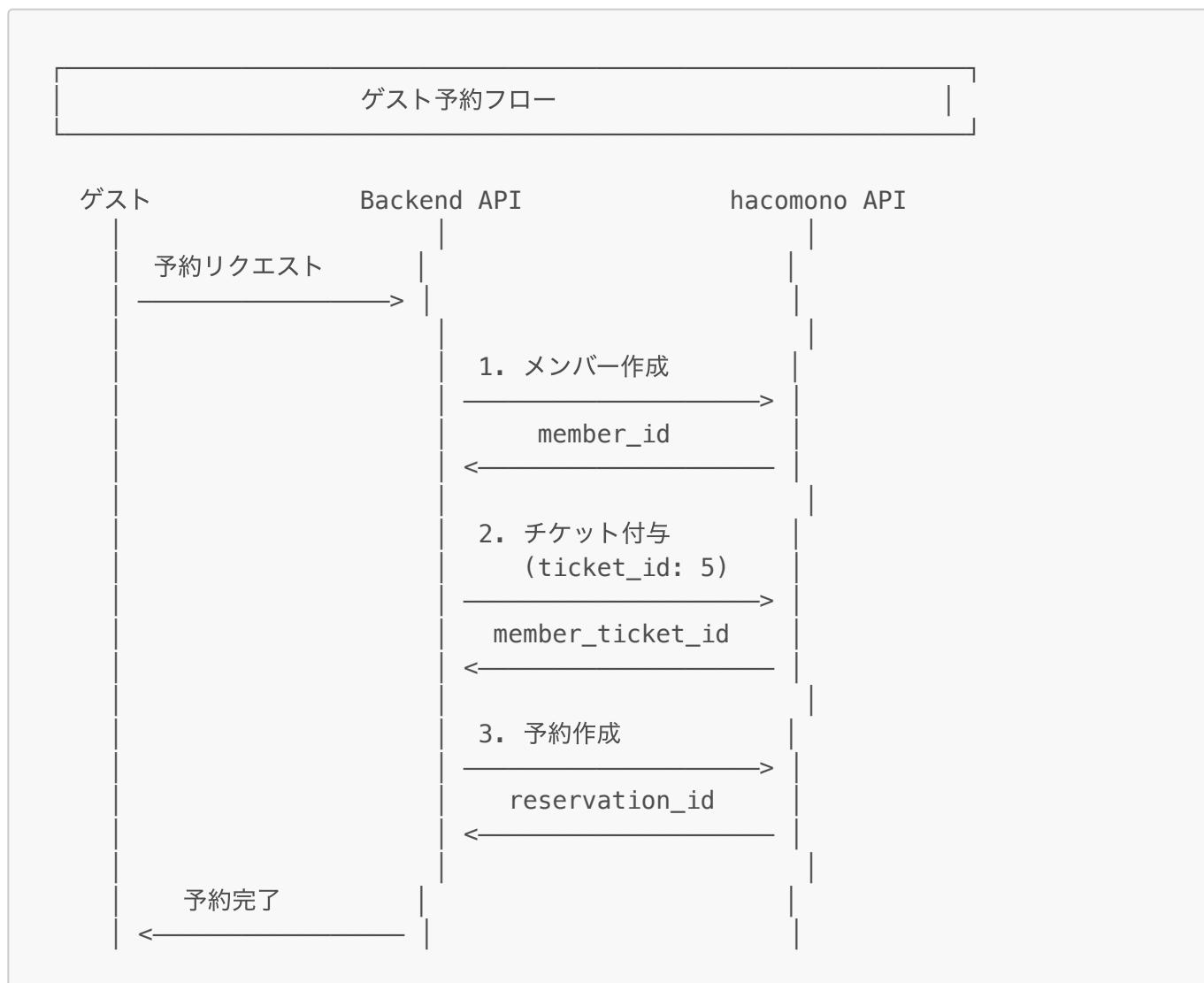
## チケット = 予約するための権利（回数券のようなもの）

- メンバーがチケットを持っていないと予約できない (RSV\_000005 エラー)
- 管理画面で事前にチケットを作成しておく必要がある
- 予約時にチケットが消費される

### 現在のチケット一覧

ID	名前	用途
1	Trial Ticket	体験用
2	One Time Ticket	1回券
3	初回体験	初回用
4	Test Ticket	テスト
5	Web予約チケット	ゲスト予約用 <input checked="" type="checkbox"/>

### ゲスト予約の自動化フロー



バックエンドでの自動処理

1. メンバー自動作成: ゲストの情報（名前、メール、電話番号）でメンバーを作成
2. チケット自動付与: ticket\_id: 5 (Web予約チケット) を1枚付与
3. 予約作成: チケットを使って予約を作成

## 実装コード例 (バックエンド)

```
def _create_guest_member(client, guest_name, guest_email, guest_phone,  
...):  
    # 1. メンバー作成  
    member_response = client.create_member({  
        "last_name": "ゲスト",  
        "first_name": "太郎",  
        "mail_address": guest_email,  
        "tel": guest_phone,  
        "plain_password": "自動生成パスワード",  
        "gender": 1,  
        "birthday": "2000-01-01",  
        "studio_id": 2  
    })  
    member_id = member_response["data"]["member"]["id"]  
  
    # 2. チケット付与  
    ticket_response = client.grant_ticket_to_member(  
        member_id,  
        ticket_id=5,  # Web予約チケット  
        num=1  
    )  
    member_ticket_id = ticket_response["data"]["member_ticket"]["id"]  
  
    return member_id, member_ticket_id
```

## テスト結果 (2025-12-12 成功)

```
# 1. メンバー作成  
作成されたメンバーID: 34  
  
# 2. チケット付与  
付与されたチケットID: 23  
  
# 3. 予約作成 (自由枠)  
{  
    "reservation": {  
        "id": 1190,  
        "member_id": 34,  
        "member_ticket_id": 23,  
        "status": 2,  
        "start_at": "2025-09-30T21:00:00+09:00",  
        "end_at": "2025-09-30T21:30:00+09:00"  
    }  
}
```

## 1. 固定枠予約（レッスン予約）

API エンドポイント

```
POST /reservation/reservations/reserve
```

必須パラメータ

パラメータ	型	説明
member_id	int	予約するメンバーのID
studio_lesson_id	int	レッスン枠のID
no	string	スペース番号 ( <code>studio_room_space.space_details[].no_label</code> の値)
member_ticket_id	int	メンバーが保有するチケットのID（チケット制の場合）

前提条件（管理画面で設定が必要）

1. 店舗（Studio）の設定

- 店舗が作成されていること
- `GET /master/studios` で確認

2. プログラム（Program）の設定

- プログラムが作成され、有効になっていること
- 店舗に紐づいていること
- `GET /master/programs` で確認

3. スタジオルーム（Studio Room）の設定

- 部屋・エリアが作成されていること
- 予約タイプ：「固定枠予約」に設定
- `GET /master/studio-rooms` で確認

4. スタジオルームスペース（Studio Room Space）の設定 △重要

- スペース（席・ベッド等）が作成されていること
- `space_details` に `no` フィールドが必要（予約時の席番号として使用）
- `GET /master/studio-room-spaces/{studio_room_id}` で確認

予約可能なスペースの条件：

`space_details` に "no" フィールドがあること

✓ 正しい例 (Pilates Spaces) :

```
space_details: [
    {"no": 1, "coord_x": 0, "coord_y": 0},
    {"no": 2, "coord_x": 1, "coord_y": 0},
    {"no": 3, "coord_x": 2, "coord_y": 0}
]
```

→ 席番号 1, 2, 3 に予約可能

✗ 間違った例 (Bookom Space 10) :

```
space_details: [
    {"type": "POSITION", "coord_x": 0, "coord_y": 0, "no_label": "1"}
]
```

→ "no" がないため予約不可！

### no と no\_label の違い:

フィールド	用途	予約時の使用
no	席番号（実データ）	<span style="color: green;">✓</span> 予約APIの no パラメータに使用
no_label	表示用ラベル	<span style="color: red;">✗</span> 予約には使えない

## ✓ 5. スタッフ (Instructor) の設定

- スタッフが登録され、有効になっていること
- GET /master/instructors で確認

## ✓ 6. レッスン枠 (Studio Lesson) の作成

- レッスン枠が作成されていること
- date : レッスン日
- start\_at / end\_at : 開始・終了時刻
- program\_id : プログラムID
- instructor\_id : 担当スタッフID
- studio\_room\_space\_id : スペースID
- published\_at : 公開日時（この日時以降に予約可能）
- GET /master/studio-lessions で確認
- POST /master/studio-lessions で作成

## ✓ 7. メンバー (Member) の作成

- 予約するメンバーが登録されていること
- 必須フィールド:
  - last\_name : 姓
  - first\_name : 名
  - mail\_address : メールアドレス
  - plain\_password : パスワード
  - gender : 性別 (1=男性, 2=女性)
  - birthday : 生年月日

- `studio_id` : 所属店舗ID
- `POST /member/members` で作成

## ✓ 8. チケットの付与（チケット制の場合）

- メンバーにチケットを付与していること
- チケットがプログラムで使用可能に設定されていること
- `POST /member/members/{member_id}/member_tickets` で付与

エラーと対処法

エラーコード	メッセージ	原因・対処
<code>CMN_000051</code>	必須パラメータが含まれていません	上記必須パラメータを確認
<code>CMN_000001</code>	エラーが発生しました	スペースの <code>space_details</code> に <code>no</code> がない可能性
<code>RSV_000005</code>	チケットが不足しています	メンバーにチケットを付与する
<code>RSV_000309</code>	営業時間外の日時を指定することはできません	レッスン枠の日時が営業時間内か確認
<code>SPACE_NO_MISSING</code>	席番号が設定されていません	スペースの設定を修正（下記参照）

スペース設定エラーの対処法

`CMN_000001` (汎用エラー) が出る場合、スペースの席設定が原因の可能性があります：

### 1. 管理画面でスペース設定を確認

- 設定 → スタジオルームスペース
- 該当スペースの編集画面を開く

### 2. 席を正しく追加

- 席を追加し、番号（no）を設定
- 例: 席1, 席2, 席3...

### 3. APIで確認

```
GET /master/studio-room-spaces/{space_id}
# space_details に "no" フィールドがあるか確認
```

## 2. 自由枠予約（choice予約）

API エンドポイント

```
POST /reservation/reservations/choice/reserve
```

## 必須パラメータ

パラメータ	型	説明
member_id	int	予約するメンバーのID
studio_room_id	int	スタジオルームのID
program_id	int	プログラムのID
start_at	string	開始日時 (yyyy-MM-dd HH:mm:ss.fff 形式)
instructor_ids	array[int]	担当スタッフIDの配列
ticket_id	int	チケットのID

前提条件（管理画面で設定が必要）

### 1. 店舗 (Studio) の設定

- 店舗が作成されていること

### 2. プログラム (Program) の設定

- プログラムが作成され、有効になっていること

### 3. スタジオルーム (Studio Room) の設定

- 部屋・エリアが作成されていること
- 予約タイプ: 「自由枠予約」に設定
- GET /master/studio-rooms で確認

### 4. スタジオルームサービス (Studio Room Service) の設定 △重要

- 自由枠予約の設定が作成されていること
- start\_date/end\_date: 予約可能期間（この期間外は予約不可）
- schedule\_nick: 予約の時間刻み（分単位、例: 30）
- GET /master/studio-room-services で確認
- POST /master/studio-room-services で作成

```
// 例: 自由枠予約設定
{
  "studio_room_id": 3,
  "name": "自由枠予約",
  "start_date": "2025-01-01",
  "end_date": "2026-03-31", // ← この期間外は予約不可！
  "schedule_nick": 30,
  "is_active": true
}
```

## ✓ 5. スタッフ (Instructor) の設定

- スタッフが登録され、有効になっていること

## ✓ 6. シフト (Shift) の作成 △重要

- 月別シフトが作成されていること
- 管理画面の「シフト管理」で12ヶ月分のシフトを追加
- GET /reservation/shifts で確認

シフトがないと → shift\_studio\_business\_hour が空 → 営業時間外エラー

## ✓ 7. 営業時間 (Shift Studio Business Hour) の設定 △重要

- シフトに対して営業時間が設定されていること
- is\_holiday: false で営業日として設定
- GET /reservation/shift\_studio\_business\_hours で確認

```
// 例: 営業時間
{
  "shift_id": 12,
  "studio_id": 2,
  "date": "2025-09-30",
  "start_at": "2025-09-30T09:00:00+09:00",
  "end_at": "2025-09-30T21:00:00+09:00",
  "is_holiday": false
}
```

## ✓ 8. スタッフシフト (Shift Instructor) の設定

- スタッフのシフトが登録されていること
- シフト時間内でないと予約不可
- 管理画面の「シフト管理」でスタッフのシフトを設定

## ✓ 9. メンバー (Member) の作成

- 予約するメンバーが登録されていること
- 固定枠と同じ必須フィールド

## ✓ 10. チケットの付与

- メンバーにチケットを付与していること
- チケットがプログラムで使用可能に設定されていること

スケジュール確認API

予約可能な時間帯を確認するには：

```
GET /reservation/reservations/choice/schedule?studio_room_id={id}&date={yyyy-MM-dd}
```

レスポンス：

- `shift_studio_business_hour`：営業時間
- `shift_instructor`：スタッフのシフト
- `reservation_assign_instructor`：予約済みスタッフ（この時間は予約不可）

エラーと対処法

エラーコード	メッセージ	原因・対処
CMN_000051	必須パラメータが含まれていません	上記必須パラメータを確認
RSV_000005	チケットが不足しています	メンバーにチケットを付与する
RSV_000308	スタッフが選択されていない、または不正	<code>instructor_ids</code> を正しく指定
RSV_000309	営業時間外の日時を指定することはできません	下記チェックリスト参照

### RSV\_000309 エラーのチェックリスト

1. シフト（月別）が存在するか？
  - 管理画面 → シフト管理 → 該当月のシフトを追加
2. 営業時間が設定されているか？
  - 管理画面 → シフト管理 → 該当日の営業時間を設定
  - `is_holiday: false` になっているか
3. スタジオルームサービスの期間内か？
  - `start_date ~ end_date` の範囲内か確認
4. スタッフのシフトが入っているか？
  - 該当時間にスタッフのシフトがあるか確認

---

### 3. 共通：メンバー作成の必須パラメータ

```
{
  "last_name": "山田",
  "first_name": "太郎",
  "mail_address": "test@example.com",
  "tel": "090-1234-5678",
  "plain_password": "SecurePass123!",
  "gender": 1,           // 1=男性, 2=女性 (0は無効)
  "birthday": "1990-01-01",
```

```
"studio_id": 2  
}
```

## 4. 実際の予約成功例

### 固定枠予約（レッスン）

```
curl -X POST  
"https://apple.admin.egw.hacomono.app/api/v2/reservation/reservations/rese  
rve" \  
-H "Authorization: Bearer $ACCESS_TOKEN" \  
-H "X-Requested-With: XMLHttpRequest" \  
-H "Content-Type: application/json" \  
-d '{  
    "member_id": 33,  
    "studio_lesson_id": 100,  
    "no": "1",  
    "member_ticket_id": 50  
}'
```

### 自由枠予約（choice）

```
curl -X POST  
"https://apple.admin.egw.hacomono.app/api/v2/reservation/reservations/choi  
ce/reserve" \  
-H "Authorization: Bearer $ACCESS_TOKEN" \  
-H "X-Requested-With: XMLHttpRequest" \  
-H "Content-Type: application/json" \  
-d '{  
    "member_id": 33,  
    "studio_room_id": 3,  
    "program_id": 2,  
    "ticket_id": 5,  
    "instructor_ids": [1],  
    "start_at": "2025-09-30 20:00:00.000"  
}'
```

## 5. 管理画面での設定チェックリスト

### 固定枠予約を使う場合

- 店舗を作成
- プログラムを作成・有効化
- スタジオルームを作成（予約タイプ：固定枠）
- スペースを作成（no\_label を設定）

- スタッフを登録
- レッスン枠を作成
- チケットを作成・プログラムに紐づけ

## 自由枠予約を使う場合

- 店舗を作成
- プログラムを作成・有効化
- スタジオルームを作成（予約タイプ：自由枠）
- **スタジオルームサービスを作成（期間を正しく設定！）**
- スタッフを登録
- **シフト（月別）を追加**
- **営業時間を設定（定休日以外）**
- **スタッフのシフトを設定**
- チケットを作成・プログラムに紐づけ

---

## 6. よくあるトラブルと解決策

Q: 「エラーが発生しました」 (CMN\_000001) が出る

A: スペースの席設定が原因の可能性が高いです：

1. スペースの **space\_details** に **no** フィールドがあるか確認

```
GET /master/studio-room-spaces/{space_id}
```

2. **no** がない場合は管理画面で設定

- 設定 → スタジオルームスペース → 該当スペース
- 席を追加 (no=1, no=2, ...)

3. バックエンドは **no** があるスペースのレッスンのみ表示

- 自動的にフィルタリングされます

Q: 「営業時間外」 エラーが出る

A: 以下を順番にチェック：

1. シフト（月別）が存在するか → 管理画面で追加
2. 営業時間が設定されているか → 管理画面で設定
3. スタジオルームサービスの期間内か → API で **end\_date** を延長
4. スタッフのシフトがあるか → 管理画面で設定

Q: 「スタッフが不正」 エラーが出る

A: **instructor\_ids** にシフトが入っているスタッフのIDを指定する。 **choice/schedule** API で **shift\_instructor** を確認。

Q: 「チケット不足」エラーが出る

A: メンバーにチケットを付与する。チケットがプログラムで使えるよう設定。

Q: 同じメンバーで予約が重複する

A: 新しいメンバーを作成するか、別の時間帯を選択する。

Q: レッスン枠が表示されない

A: スペースの席設定（**no**フィールド）がないレッスンは自動的にフィルタされます：

#### 1. APIでスペースを確認

```
GET /master/studio-room-spaces
# space_details に "no" があるスペースのみ予約可能
```

#### 2. 予約可能なスペースを使用したレッスン枠を作成