

乗換案内 Biz SDK 仕様書

オフピーク定期対応

ジョルダン株式会社
2023/02/21

概要

乗換案内 Biz SDK は JavaScript を用いて乗換案内 Biz の画面の操作及び結果を取得する開発ツールです。

この文書では、オフピーク定期に関連する乗換案内 Biz の画面及び JavaScript 関数の仕様について記載します。

目次

概要.....	2
画面仕様.....	4
関数仕様.....	5

画面仕様

Copyright © 2022 Jorudan Co.,Ltd. All Rights Reserved.

乗換案内 Biz
- 10ライセンス版 -

三鷹 ~ 新宿御苑前 2023年3月18日

経路	評価	時間	運賃 / IC	乗換	距離
1	早 安 楽 飛 登	27分	388円	1回	14.8km
2	早 安 楽 飛 登	31分	476円	1回	19.4km
3	早 安 楽 飛 登	30分	356円	1回	14.0km
4	早 安 楽 飛 登	1時間02分	410円	2回	27.6km

経路 1

通勤定期代 / オフピーク定期		1ヶ月	3ヶ月	6ヶ月	払戻
JR	中央線 三鷹 ~ 新宿	6,580円	18,760円	31,620円	※
M	東京メトロ丸ノ内線 新宿 ~ 新宿御苑前	6,260円	17,850円	33,810円	※
合計		12,840円	36,610円	65,430円	

- ・ オフピーク定期については、定期代画面にて金額を表示します
- ・ 「通勤定期代 / オフピーク定期」の項目をクリックすることでそれぞれの定期代表示を切り替えます
- ・ オフピーク定期の金額が設定されている場合は異なる背景色（緑色）で表示されます
- ・ オフピーク定期が設定されていない区間については通常の通勤定期代を表示します
- ・ オフピーク定期の払い戻し計算については現時点で非対応となります
- ・ 2023年3月版の乗換案内 Biz の初期状態ではオフピーク定期は非表示となります。
表示させるには後述の設定関数を呼び出して表示設定を変更させるか、
インストール先の Client/jrdweb/jrdusr.js ファイル内、jrd.sdk.initCustomize 内に設定関数を記述してください。
- ・ オフピーク定期を利用した定期精算（ピーク時間帯における定期精算）を行う場合は後述の設定関数を呼び出して定期精算モードを変更し、オフピーク対応の定期区間登録関数にてオフピーク定期として登録してください。
- ・ 実際のピーク時間帯の判定は改札の通過時に行われますが、乗換案内 Biz で扱っている列車の発着時刻からは改札の通過時刻の判断が出来ないため、検索時刻による自動的なピーク時間帯の判定は行いません。
- ・ 初期状態では従来通りの定期精算処理を行います。

関数仕様

○オフピーク定期表示設定

- ・関数名

JrdSetOffpeakTeikiMode(n)

- ・説明

定期代画面におけるオフピーク定期の表示状態を設定します

- ・パラメータ

n = 0 : 非表示 (デフォルト)

1 : 通勤定期代を初期表示

2 : オフピーク定期を初期表示

- ・戻り値

セットされたオフピーク定期表示の設定値

○定期表示状態の取得

- ・関数名

JrdGetNtResultDisp()

- ・説明

定期代画面の表示状態を取得します

- ・パラメータ

無し

- ・戻り値

1 : 通勤定期が表示されている

2 : オフピーク定期が表示されている

-1 : 検索結果が存在しない

○オフピーク定期存在チェック

・関数名

JrdGetNtOffpeakExist1(rno, pno)

JrdGetNtOffpeakExist3(rno, pno)

JrdGetNtOffpeakExist6(rno, pno)

・説明

指定された経路番号、路線番号における定期代にオフピーク定期が存在するかを取得します。

関数名末尾の 1, 3, 6 はそれぞれ 1 か月定期、3 か月定期、6 か月定期を意味します。

・パラメータ

rno : 経路番号

pno : 路線番号

※経路番号、路線番号は通常の定期代取得と同じです

・戻り値

0 : オフピーク定期が存在しない

1 : オフピーク定期が存在する

-1 : 結果が存在しない

○オフピーク定期代取得

・関数名

JrdGetNtOffpeak1(rno, pno)

JrdGetNtOffpeak3(rno, pno)

JrdGetNtOffpeak6(rno, pno)

・説明

指定された経路番号、路線番号におけるオフピーク定期代を取得します。

関数名末尾の 1, 3, 6 はそれぞれ 1 か月定期、3 か月定期、6 か月定期を意味します。

・パラメータ

rno : 経路番号

pno : 路線番号

※経路番号、路線番号は通常の定期代取得と同じです

・戻り値

定期代（オフピーク定期が存在しない場合は通常の通勤定期代を返します）

-1：結果が存在しない

○オフピーク定期の運賃 ID 取得

・関数名

JrdGetNtOffpeakId1(rno, pno)

JrdGetNtOffpeakId3(rno, pno)

JrdGetNtOffpeakId6(rno, pno)

・説明

指定された経路番号、路線番号におけるオフピーク定期の運賃 ID を取得します。

ID が同じ区間は金額が通算されていることを表します。

関数名末尾の 1, 3, 6 はそれぞれ 1 か月定期、3 か月定期、6 か月定期を意味します。

・パラメータ

rno：経路番号

pno：路線番号

※経路番号、路線番号は通常の定期代取得と同じです

・戻り値

オフピーク定期の運賃 ID

無効な場合 -1

○オフピーク定期代合計額取得

・関数名

JrdGetNtOffpeakTotal1(rno)

JrdGetNtOffpeakTotal3(rno)

JrdGetNtOffpeakTotal6(rno)

・説明

指定された経路番号におけるオフピーク定期代の合計額を取得します。

関数名末尾の 1, 3, 6 はそれぞれ 1 か月定期、3 か月定期、6 か月定期を意味します。
オフピーク定期が存在しない場合は通常の定期代を含めた金額となります。

・パラメータ

rno : 経路番号

※経路番号は通常の定期代取得と同じです

・戻り値

定期代

-1 : 結果が存在しない

○定期区間設定 (オフピーク定期対応)

・関数名

JrdAddTeikiEx(rosen, from, to, offpeak)

・説明

定期区間を登録します。

オフピーク定期として登録する場合は offpeak の値を 1 にしてください。

・パラメータ

rosen : 路線名

from : 発名称

to : 着名称

offpeak : オフピークフラグ (0:通常の定期、1:オフピーク定期)

・戻り値

なし

○定期精算モード設定

・関数名

JrdSetTeikiSeisanMode(n)

・説明

定期精算処理を行う際のモードを設定します。

ピーク時間帯モードの場合、オフピーク定期として登録されている定期区間は精算されません。

※オフピーク定期が実際に存在しているかどうかの自動判定は行いません。

※時刻検索に伴うピーク時間帯の自動判定は行いません。

- ・パラメータ

n = 0 : 通常の定期精算 (デフォルト)

1 : ピーク時間帯モード

- ・戻り値

セットされた定期精算モードの値

○結果の定期精算モード取得

- ・関数名

JrdGetNsSeisanMode()

- ・説明

定期精算結果がどの定期精算モードで計算されたかを取得します

- ・パラメータ

無し

- ・戻り値

0 : 通常の定期精算 (デフォルト)

1 : ピーク時間帯モード