

# **JSON Api pro WebCall**

Kadlec elektronika s.r.o.

verze 4.2.0, 08.01.2025

# Obsah

JSON rozhraní - definice	1
1 Popis způsobů volání rozhraní JSON	3
1.1 apiKey	3
1.2 Volání bez URL parametru	3
1.3 Volání s parametrem method	3
1.4 Návrátové hodnoty	3
2 Seznam a popis metod s volacím a návratovým objektem	4
2.1 Objednávání	4
2.2 Zrušení Objednávky	15
2.3 Monitoring	17
2.4 Ostatní	23
2.6 Šablona chybové zprávy	34

## 1 Popis způsobů volání rozhraní JSON

Rozhraní JSON se nachází v souboru `json.php`. Každé volání tedy bude probíhat formou požadavku na adresu `<instalace>/json.php`.

### 1.1 apiKey

Od verze API 4.0.0 se veškeré metody volají s `apiKey`, které je jedinečné pro každou pobočku a obdržíte jej po zakoupení licence.

```
{
    "poboID":101,
    "cinnostID":"9",
    "email":"email@email.com",
    "den":"2020-08-26",
    "pin":"2136",
    "apiKey":"f562de4bf6cf722ae433befdb3a1b0f06de3eeac"
}
```

### 1.2 Volání bez URL parametru

První způsob volání probíhá přímo zasláním požadavku se slovesem (metodou) *POST* na soubor `JSON.php`. Pokud nespecifikujeme v url parametr *method*, skript se pokusí vyčíst název volané metody z těla zasláního objektu. Očekávaná zpráva má následující tvar:

```
query={
    "methodName": <název volané metody>,
    "params": {
        <název parametru metody>: <hodnota>
    }
}
```

Veškeré informace o volané funkci jsou obsaženy v parametru `query`. Ten obsahuje dva atributy - „methodName“ s názvem požadované funkce a dále „params“ s jejími parametry.

### 1.3 Volání s parametrem method

Druhá varianta volání používá požadavek se slovesem (metodou) *POST* na soubor `JSON.php`, ale

zaslaný požadavek navíc obsahuje v url parametr *method*. Tento parametr se používá ke stanovení volané metody, jeho hodnota je přímo použita jako požadovaný název. Veškeré parametry jsou předány v těle zprávy v podobě JSON objektu. Očekávané tělo má následující tvar:

```
{
    "<název parametru>":"hodnota",
    "<název parametru>":"hodnota",
    ...
}
```

celý objekt je zde serializován do dvojic. Název každého atributu je použit jako odpovídající název parametru volané funkce a hodnota atributu je předána jako hodnota tohoto parametru.

V dokumentaci budeme rozebírat podrobně první variantu. Druhá varianta jde z ní odvodit tak, že název funkce (zvýrazněno zeleně) zadáme do url parametru *method* a objekt "params" pošleme jako tělo zprávy, bez názvu POST proměnné.

## 1.4 Návrátové hodnoty

V případě, že funkce bude správně převedena, její návratovou hodnotou bude objekt serializovaný ve formátu JSON, který obsahuje veškerá vrácená data. V opačném případě bude návratovou hodnotou serializovaná chybová zpráva popsána níže.

## 2 Seznam a popis metod s volacím a návratovým objektem

### 2.1 Objednávání

`obj_dej_cinnost(long poboID);`

#### UPDATE

Získání všech činností zadané pobočky.

URL dotaz: `json.php?method=obj_dej_cinnost`

Tělo URL dotazu:

```
{
    "poboID":    <long>
}
```

Dotaz objektem:

```
{
    "methodName": "obj_dej_cinnost",
    "params": {
        "poboID":    <long>
    }
}
```

poboID: celočíselná identifikace pobočky,

Odpověď:

```
{
    "return": [
        {
            "cinnostID":    <long>,
            "nazev":        <string>,
            "pozn":         <string>,
            "id_callxx":    <string>,

```

```

        "nadrzena_skupina_h":    <long>,
        "nadrzena_skupina_u":    <long>,
        "informace":            <string>
        "textTop":               <string>
        "textBottom":           <string>
        "upload":                <bool>
        "objNumberOfDays":       <int>
    }
    ...
]
}

```

**cinnostID:** identifikace činnosti,  
**nazev:** název činnosti (z CALL250/WebCall),  
**pozn:** doplňující text (souvisí s WC\_OBJEDNAVKA.POZN – kam je ukládán text od klienta, tato položka je pak text, který komentuje uživatelskou položku ve formuláři, v době, kdy text poznámky zadává),  
**id\_callxx:** identifikace v Call250V, pořadí činnosti od 0 zvýšení o 1 s předponou A, např. A1, A2, ...  
**nadrzena\_skupina\_h:** identifikace skupiny, do které je činnost zařazena pro účely hierarchie činností; je-li obsahem 0, činnost není zařazena do žádné skupiny a znamená to, že je uváděna na základní úrovni v hierarchii skupin činností,  
**nadrzena\_skupina\_u:** identifikace skupiny, do které je činnost zařazena pro účely sdruženého objednávání; je-li obsahem 0, činnost není zařazena do žádné skupiny a znamená to, že je uváděna na základní úrovni v hierarchii skupin činností,  
**informace:** dodatečné informace o činnosti pobočky – jedná se o odkaz typu URI, který je využíván při zobrazování seznamu činností uživatelům portálu,  
**textTop:** text nad polem uploadu,  
**textBottom:** text pod polem uploadu,  
**upload:** aktivní / neaktivní upload u činnosti,  
**objNumberOfDays:** minimální počet dnů od kdy se klient může objednat

**obj\_dej\_skupina**(long poboID)

Získání všech skupin činností zadané pobočky. Činnosti mohou být seskupovány hierarchicky do skupin, které jsou pojmenovány.

URL dotaz: json.php?method=obj\_dej\_skupina

Tělo URL dotazu:

```

{
    "poboID":    <long>
}

```

Dotaz objektem:

```

{
    "methodName": "obj_dej_skupina",
    "params": {
        "poboID":    <long>
    }
}

```

poboID: celočíselná identifikace pobočky,

Odpověď:

```
{
  "return": [
    {
      "skupinaID":          <long>,
      "nazev":              <string>,
      "nadrazena_skupina_h": <long>,

      "informace":          <string>,
      "skupina_typ":        <string>,
    },
    ...
  ]
}
```

skupinaID: identifikace skupiny činnosti,

nazev: název skupiny činností,

nadrazena\_skupina\_h: identifikace nadřazené skupiny činnosti,

informace: dodatečné informace o skupině činností pobočky – jedná se o odkaz typu URI, který je využíván při zobrazování seznamu činností uživatelům portálu,

skupina\_typ: možné hodnoty „U“ nebo „H“ podle toho, zda jde o skupinu činností pro účely sdruženého objednávání, nebo pro účely hierarchie činností.

**obj\_dej\_den**(long poboID, long cinnostID);

Získání kalendářních dnů s vyjádřením stavu objednávání na dané pobočce na danou činnost v rozsahu objednávání dle parametrů `O_DNY_DOPREDU_MIN` a `O_DNY_DOPREDU_MAX`,

URL dotaz: `json.php?method=obj_dej_den`

Tělo URL dotazu:

```
{
  "poboID":          <long>,
  "cinnostID":       <long>
}
```

Dotaz objektem:

```
{
  "methodName": "obj_dej_den",
  "params": {
    "poboID":          <long>,
    "cinnostID":       <long>
  }
}
```

poboID: celočíselná identifikace pobočky,

cinnostID: celočíselná identifikace činnosti na pobočce,

Odpověď:

```
{
  "return": [
    {
      "dt_datum":      <string>,
      "stav":          <int>
    },
    ...
  ]
}
```

dt\_datum: datum ve formátu RRRR-MM-DD,  
stav: vyjádření k možnosti se objednávat na daný den, může být:  
0:objednání na tento den je zakázáno,  
1:je možné se objednávat,  
2:termín obsazen.

**obj\_dej\_cas**(long poboID, long cinnostID, string den);

Získání všech časů, na které se ze lze daný den na danou činnost u dané pobočky objednávat.

URL dotaz: json.php?method=obj\_dej\_cas

Tělo URL dotazu:

```
{
  "poboID":      <long>,
  "cinnostID":   <long>,
  "den":         <string>
}
```

Dotaz objektem:

```
{
  "methodName": "obj_dej_cas",
  "params": {
    "poboID":      <long>,
    "cinnostID":   <long>,
    "den":         <string>
  }
}
```

poboID: celočíselná identifikace pobočky,  
cinnostID: celočíselná identifikace činnosti na pobočce,  
den: je zadáván ve formátu RRRR-MM-DD.

Odpověď:

```
{
  "return": [
    {
      "cas":      <string>,
      "stav":     <int>
    },
    ...
  ]
}
```

cas(string) : čas který je uveden pro objednávání ve formátu HH:MM,  
stav : stav pro použití uvedeného času, možné hodnoty:  
0: čas pro objednávání je již obsazen,  
1: čas pro objednávání je volný,

```
obj_uloz_objednavka(long poboID, long cinnostID, string email, string jmeno,  
string den, string cas, string tel, int sms, string pozn, string klient);
```

Rezervace objednávky na pobočce na danou činnost, den, čas se zadáním emailové adresy, jména, příjmení, telefonu,...

URL dotaz: json.php?method=obj\_uloz\_objednavka

Tělo URL dotazu:

```
{  
    "poboID":      <long>,  
    "cinnostID":   <long>,  
    "email":       <string>,  
    "jmeno":       <string>,  
    "den":         <string>,  
    "cas":         <string>,  
    "tel":         <string>,  
    "sms":         <int>,  
    "pozn":        <string>,  
    "klient":      <string>  
}
```

Dotaz objektem:

```
{  
    "methodName": "obj_uloz_objednavka",  
    "params": {  
        "poboID":      <long>,  
        "cinnostID":   <long>,  
        "email":       <string>,  
        "jmeno":       <string>,  
        "den":         <string>,  
        "cas":         <string>,  
        "tel":         <string>,  
        "sms":         <int>,  
        "pozn":        <string>,  
        "klient":      <string>  
    }  
}
```

poboID: celočíselná identifikace pobočky,  
cinnostID: celočíselná identifikace činnosti na pobočce,  
email: adresa elektronické pošty klienta,  
jmeno: jméno a příjmení, popř. jiný identifikátor klienta (název firmy),  
den: je zadáván ve formátu RRRR-MM-DD,  
cas: je zadáván ve formátu HH:MM,

tel: je zadáván v běžně používaném formátu 9místného čísla s nebo bez mezinárodní předvolby,

sms: slouží k oznámení, zda je požadováno oznámení o vytvoření objednávky také prostřednictvím textové zprávy SMS – v případě použití parametru se stává povinným také parametr tel;

pozn: slouží pro přenos textové informace do systému Call250V

klient: klient je uváděn právě pokud je objednávka ukládána pro registrovaného klienta

#### Odpověď:

```
{
  "return": {
    "jmeno":      <string>,
    "email":      <string>,
    "tel":        <string>,
    "datum":      <string>,
    "cas":        <string>,
    "pin":        <int>,
    "text_potvrzeni": <string>,
    "pobocka":    <string>,
    "nazev_cinnosti": <string>,
    "text_ew":    <string>,
    "pozn":       <string>
  }
}
```

jmeno: jméno a příjmení, popř. jiný identifikátor klienta (název firmy),

email: adresa elektronické pošty klienta,

tel: telefon klienta,

datum: datum termínu objednávky ve formátu RRRR-MM-DD,

cas: čas termínu objednávky ve formátu HH:MM,

pin: PIN, který klient zadá po příchodu na pobočku,

text\_potvrzeni: potvrzovací text nastavitelný v administraci,

pobocka: název pobočky,

nazev\_cinnosti: název činnosti, na kterou byla objednávka vytvořena,

text\_ew: text specifický pro činnost, který se používá jako substituce textu %acttext% u odesílání potvrzovacích emailů nebo výpisu potvrzovacích webových stránek,

pozn: poznámka,

`obj_parametry`(int poboID, string nazev);

Vrací hodnotu nastavení parametru podle názvu parametru v nazev dané pobočky poboID; pokud nazev není použit, metoda vrací všechny parametry požadované pobočky, které jsou potřebné pro objednávání;.

URL dotaz: json.php?method=obj\_parametry

Tělo URL dotazu:

```
{
  "poboID":    <long>,
  "nazev":     <string>
}
```



```
}
```

Dotaz objektem:

```
{
    "methodName": "obj_dej_den",
    "params": {
        "poboID":    <long>,
        "nazev":     <string>
    }
}
```

poboID: je celočíselná identifikace pobočky,

nazev: je nepovinný název parametru,

Odpověď:

```
{
    "return": [
        {
            "nazev":    <string>,
            "obsah":    <string>
        },
        ...
    ]
}
```

nazev : název parametru,

obsah : obsah parametru, nastavení pobočky,

```
obj_dej_objednavka(string klientID , string dt_od, string dt_do);
```

Vrací všechny objednávky registrovaného klienta v rozmezí časového rozpětí podle zadaného data.

URL dotaz: json.php?method=obj\_dej\_objednavka

Tělo URL dotazu:

```
{
    "klientID":    <string>,
    "dt_od":       <string>,
    "dt_do":       <string>
}
```

Dotaz objektem:

```
{
    "methodName": "obj_dej_objednavka",
    "params": {
        "klientID":    <string>,
        "dt_od":       <string>,
        "dt_do":       <string>
    }
}
```

dt\_od: datum od včetně, formát RRRR-MM-DD,

dt\_do: datum do včetně, formát RRRR-MM-DD.

Nejsou-li parametry dt\_od a dt\_do vyplněny, vrací všechny objednávky klienta. Je-li použit pouze

jeden z parametrů dt\_od nebo dt\_do, časové rozmezí je takto ohraničeno buďto shora nebo zdola.

Odpověď:

```
{
  "return": [
    {
      "dt_nakdy": <string>,
      "dt_odkdy": <string>,
      "jmeno":      <string>,
      "pin":        <int>,
      "email":      <string>,
      "tel":        <string>,
      "stav":       <int>,
      "dt_probehla": <string>,
      "dt_zrusena": <string>,
      "pozn":       <string>,
      "obecny_priznak": <string>,
      "objednavkaID": <long>,
      "klientID": <string>,
      "poboID":    <long>,
      "poboNazev": <string>,
      "cinnostID": <long>,
      "cinnostNazev": <string>,
      "id_callxx": <string>
    },
    ...
  ]
}
```

dt\_nakdy: datum a čas, kdy je objednávka obsloužena ve formátu RRRR-MM-DD HH:MM,

dt\_odkdy: datum a čas vytvoření objednávky ve formátu RRRR-MM-DD HH:MM,

jmeno: jméno a příjmení, popř. jiný identifikátor klienta (název firmy),

pin: PIN, který klient zadá po příchodu na pobočku,

email: email objednávky,

tel: telefonní číslo objednávky,

stav: stav objednávky, v návrhu jsou tyto hodnoty:

0 – je platná a očekává se, že proběhne,

2 – nebyla potvrzena klientem emailem do daného časového limitu,

3 -- zrušena klientem,

4 -- čeká na potvrzení emailem v probíhajícím časovém limitu,

5 -- byla zrušena správcem nebo manažerem,

6 -- zrušena, ale nepotvrzena klientem,

7 -- objednávka propadla, klient se nedostavil,

8 – objednávka proběhla, klient se zpozdil,

9 – objednávka proběhla a byla zavedena do archivu

dt\_probehla: datum a čas, kdy skutečně byla objednávka standardně realizována, pokud se tak stalo ve formátu RRRR-MM-DD HH:MM,

dt\_zrusena: datum a čas, kdy byla objednávka zrušena (klientem) ve formátu RRRR-MM-DD HH:MM,

pozn: další doprovodný text,

obecny\_priznak: pole obecných vlastností objednávky,

objednavkaID: jednoznačná číselná identifikace objednávky v databázi,  
 klientID: jednoznačná identifikace registrovaného klienta, jedná-li se o objednávku  
 neregistrovaného klienta, obsahuje 0;  
 poboID: jednoznačný identifikátor pobočky,  
 poboNazev: název pobočky,  
 cinnostID: jednoznačný identifikátor činnosti,  
 cinnostNazev: název činnosti,  
 id\_callxx: identifikace v Call250V, pořadí činnosti od 0 zvýšení o 1 s předponou A,  
 např. A1, A2, ...

`obj_dej_klient_stav(string klientID);`

Vrací obecné doplňkové vlastnosti registrovaného klienta s identifikací klientID.

URL dotaz: `json.php?method=obj_dej_klient_stav`

Tělo URL dotazu:

```
{
    "klientID":      <string>
}
```

Dotaz objektem:

```
{
    "methodName": "obj_dej_klient_stav",
    "params": {
        "klientID":      <string>
    }
}
```

Odpověď:

```
{
    "return": {
        "obecny_priznak":      <string>
    }
}
```

obecny\_priznak: obsahuje doplňkové atributy klienta, jedná se o 20 znakový řetězec, je-li  
 v prvním bajtu obsažen znak B, je klient blokován pro vytváření nových  
 objednávek;

`obj_get_status(long poboID, long cinnostID, string email, string den, string pin);`

Vrací stav objednávky.

URL dotaz: `json.php?method=obj_get_status`

Tělo URL dotazu:

```
{
    "poboID":      <long> ,
    "email":      <string>,
    "pin":      <string>,
    "den":      <string>,
    "cinnostID": <long>
```

```
}
```

Dotaz objektem:

```
{
    "methodName": "obj_get_status",
    "params": {
        "poboID":          <long> ,
        "email":           <string>,
        "pin":             <string>,
        "den":             <string>,
        "cinnostID": <long>
    }
}
```

Odpověď:

```
{
    "return": {
        "status": <integer>
    }
}
```

status:

- 0 - očekává se, že proběhne, pokud byl požadavek parametrem, byla řádně potvrzena (email),
- 1 - neobsazeno,
- 2 - nebyla vůbec potvrzena klientem (z emailu) v průběhu potvrzovacího intervalu,
- 3 - zrušena klientem,
- 4 - nebyla zatím potvrzena (email) při vytvoření (čeká na potvrzení email),
- 5 - zrušena správcem nebo manažerem,
- 6 - čeká na potvrzení pro zrušení email,
- 7 - klient se nedostavil na pobočku,
- 8 - proběhla standardně, ale s opožděně zadaným PINem,
- 9 - proběhla standardně, je v archívu

## 2.2 Zrušení Objednávky

`obj_zrus_objednavka`(long poboID, long cinnostID, string email, string den, string pin, string klientID, bool forceDisableConfirm);

Zrušení objednávky na dané pobočce dané činnosti se zadáním emailové adresy, data a PINu.

URL dotaz: `json.php?method=obj_zrus_objednavka`

**UPDATE:** den a cinnostID již nejsou povinné parametry, lze místo nich zadat -1 což znamená že nebudou kontrolovány.

Tělo URL dotazu:

```
{
    "poboID":          <long> ,
    "email":           <string>,
    "pin":             <string>,
    "den":             <string>|<int>,
    "cinnostID":       <long>,
    "klientID":        <string>,
    "forceDisableConfirm": <boolean>
}
```

Dotaz objektem:

```
{
    "methodName": "obj_zrus_objednavka",
    "params": {
        "poboID":          <long> ,
        "cinnostID": <long>,
        "email":           <string>,
        "den":             <string>,
        "pin":             <string>,
        "klientID":        <string>,
        "forceDisableConfirm": <boolean>
    }
}
```

poboID:	celočíslná identifikace pobočky,
cinnostID:	jednoznačný identifikátor činnosti, nebo -1,
email:	adresa, která byla zadána při vytvoření objednávky,
den:	datum termínu objednávky ve formátu RRRR-MM-DD nebo -1,
pin:	PIN, který byl vygenerován k objednávce,
klient:	případná identifikace registrovaného klienta.

Odpověď:

```
{
    "return": {
        "jmeno":          <string>,
        "email":          <string>,
        "tel":            <string>,
        "datum":          <string>,
    }
}
```

```

        "cas":                <string>,
        "pin":                <int>,
        "text_potvrzeni":    <string>,
        "pobocka":          <string>,
        "cinnost":          <string>
    }
}

```

jmeno: jméno a příjmení, popř. jiný identifikátor klienta (název firmy),  
 email: adresa elektronické pošty klienta,  
 tel: telefon klienta,  
 datum: datum termínu objednávky ve formátu RRRR-MM-DD,  
 cas: čas termínu objednávky ve formátu HH:MM,  
 pin: PIN, který klient zadá po příchodu na pobočku,  
 text\_potvrzeni: potvrzovací text nastavitelný v administraci,  
 pobocka: název pobočky,  
 cinnost: název činnosti, na kterou byla objednávka vytvořena.  
 forceDisableConfirm: pokud je hodnota TRUE nebude odeslán potvrzovací email.

```
obj_zrus_objednavka_by_hash(string hash, bool forceDisableConfirm);
```

Zrušení objednávky podle hash.

URL dotaz: `json.php?method=obj_zrus_objednavka_by_hash`

Tělo URL dotazu:

```

{
    "hash":                <string>,
    "forceDisableConfirm": <boolean>
}

```

Dotaz objektem:

```

{
    "methodName": "obj_zrus_objednavka_by_hash",
    "params": {
        "hash":                <string> ,
        "forceDisableConfirm": <boolean>
    }
}

```

hash: identifikace objednávky vložená do URL substitucí `%hash_d%`,

Odpověď:

```

{
    "return": {
        "jmeno":                <string>,
        "email":                <string>,
        "tel":                  <string>,
        "datum":                <string>,
        "cas":                  <string>,
        "pin":                  <int>,
        "text_potvrzeni":        <string>,
    }
}

```

```

        "pobočka":      <string>,
        "cinnost":     <string>
    }
}

```

jmeno: jméno a příjmení, popř. jiný identifikátor klienta (název firmy),  
 email: adresa elektronické pošty klienta,  
 tel: telefon klienta,  
 datum: datum termínu objednávky ve formátu RRRR-MM-DD,  
 cas: čas termínu objednávky ve formátu HH:MM,  
 pin: PIN, který klient zadá po příchodu na pobočku,  
 text\_potvrzeni: potvrzovací text nastavitelný v administraci,  
 pobočka: název pobočky,  
 cinnost: název činnosti, na kterou byla objednávka vytvořena.  
 forceDisableConfirm: pokud je hodnota TRUE nebude odeslán potvrzovací email.

## 2.3 Upload příloh

### UPDATE

```
obj_att_upload(objId, hs);
```

Upload souboru k objednávce.

URL dotaz: `json.php?method=obj_att_upload&objId=[int]&hs[string]`

**Tato metoda je výjimka**, parametry se zadávají v rámci URL, POST je využit pro upload. Lze uploadovat více souborů najednou.

objId: celočíselná identifikace objednávky,  
 hs: hash objednávky

**Oba parametry vrací metoda `obj_uloz_objednavka` která musí být použita před uploadem.**

Příklad volání metody příklad URL a obsahu POST:

```

POST
http://www.domena.cz/json.php?method=obj_att_upload&objId=6899&hs=3ff16d359a1c81da4a807179eafdef06b21c7e7 HTTP/1.1
Host: localhost
Cache-Control: no-cache
Postman-Token: 4a4702f3-c291-c41e-a7f0-f8d7c014bcc6
Content-Type: multipart/form-data; boundary=----WebKitFormBoundary7MA4YWxkTrZu0gW

-----WebKitFormBoundary7MA4YWxkTrZu0gW
Content-Disposition: form-data; name="file1"; filename="tabulka.docx"
Content-Type: application/vnd.openxmlformats-officedocument.wordprocessingml.document

-----WebKitFormBoundary7MA4YWxkTrZu0gW
Content-Disposition: form-data; name="file2"; filename="obrazek.jpg"

```

Content-Type: image/jpeg

-----WebKitFormBoundary7MA4YWxkTrZu0gW--

## 2.4 Monitoring

`mon_dej_cinnost(long poboID);`

Vrací seznam činností na pobočce, které jsou v daném okamžiku obsluhovány.

URL dotaz: `json.php?method=mon_dej_cinnost`

Tělo URL dotazu:

```
{
    "poboID":    <long>
}
```

Dotaz objektem:

```
{
    "methodName": "mon_dej_cinnost",
    "params": {
        "poboID":    <long>
    }
}
```

poboID:                      celočíselná identifikace pobočky,

Odpověď:

```
{
    "return": [
        {
            "cinnostID":    <long>,
            "nazev":        <string>,
            "fronta":        <int>,
            "pocet_prepazek":    <int>,
            "doba_cekani_odhad":    <int>,
            "m_poradi_aktivni":    <int>,
            "doba_cekani":        <int>
        }
        ...
    ]
}
```

cinnostID:    identifikace činnosti,

nazev:                      název činnosti,

fronta:                      délka fronty na činnost na pobočce,

pocet\_prepazek:    počet obsluhujících přepážek.

doba\_cekani\_odhad: odhad doby čekání na činnost v počtu minut,

m\_poradi\_aktivni: identifikace, zda je činnost pro monitoring aktivní: ano/ne,

doba\_cekani:                skutečná doba čekání na činnost v počtu minut.



`mon_dej_prepazka(long poboID);`

Vrací seznam přepážek, které v současném okamžiku na pobočce obsluhují.

URL dotaz: `json.php?method=mon_dej_prepazka`

Tělo URL dotazu:

```
{
    "poboID":    <long>
}
```

Dotaz objektem:

```
{
    "methodName": "mon_dej_prepazka",
    "params": {
        "poboID":    <long>
    }
}
```

`poboID`: celočíslná identifikace pobočky,

Odpověď:

```
{
    "return": [
        {
            "prepazkaID":    <long>,
            "nazev":         <string>,
            "stav":          <int>,
            "obsluha": <int>,
            "text":          <string>,
            "cinnosti":      <string>
        }
        ...
    ]
}
```

`prepazkaID`: identifikace přepážky,

`nazev`: název přepážky,

`stav`: stav přepážky, může nabývat následujících hodnot:

10000 – přepážka je uzavřena, ale obsluha je přítomna,

10001 – přepážka je nepřihlášena, obsluha není přítomna,

10002 – přepážka je přihlášena, obsluha je přítomna,

1 ... 9999 – přepážka obsluhuje klienta s daným číslem v uvedeném rozpětí,

`obsluha`: obsluha přepážky,

`text`: doprovodný text o stavu přepážky,

`cinnosti`: seznam činností, které jsou obsluhovány na přepážce (jsou uvedeny identifikace činností `cinnostID` oddělené čárkou),

```
mon_dej_poradi(long cinnostID, string poradove_cislo);
```

Vrací pořadí klienta ve frontě na činnost v případě, že jeho pořadové číslo z parametru `poradove_cislo` nalezne v odpovídající frontě na činnost podle parametru `cinnostID`.

URL dotaz: `json.php?method=mon_dej_poradi`

Tělo URL dotazu:

```
{
    "cinnostID": <long>,
    "poradove_cislo": <string>
}
```

Dotaz objektem:

```
{
    "methodName": "mon_dej_poradi",
    "params": {
        "cinnostID": <long>,
        "poradove_cislo": <string>
    }
}
```

`cinnostID`: identifikace činnosti,

`poradove_cislo`: vytištěné pořadové číslo na lístku z vyvolávacího systému

Odpověď:

```
{
    "return":
    {
        "poradi": <int>,
        "nazev": <string>
    }
}
```

`poradi`: pořadí klienta ve frontě na činnost,

`nazev`: název fronty na činnost.

```
mon_dej_doba(long poboID);
```

Vrací informace o době (intervalech) monitoringu pro činnosti na pobočce dle `poboID`.

URL dotaz: `json.php?method=mon_dej_doba`

Tělo URL dotazu:

```
{
    "poboID": <long>
}
```

Dotaz objektem:

```
{
    "methodName": "mon_dej_doba",
    "params": {
        "poboID": <long>
    }
}
```

`poboID`: celočíselná identifikace pobočky,

Odpověď:

```
{
  "return": [
    {
      "den":          <int>,
      "cas":          <string>,
      "cinnostID":    <long>,
      "nazev":        <string>,
      "id_callxx":    <string>
    }
    ...
  ]
}
```

den : den v týdnu, obsahuje číselnou identifikaci dne: 1 – pondělí, 2 – úterý ... 7 – neděle,  
cas : sekvence časových intervalů; jednotlivé intervaly jsou odděleny čárkou a obsahují 2 časové údaje (začátek a konec intervalu) ve formátu HH:MM oddělené pomlčkou, např. 08:30-11:00,12:00-16:00,  
cinnostID : identifikace činnosti na pobočce, ke které je časový interval vztažen,  
nazev : název činnosti na pobočce, ke které je časový interval vztažen,  
id\_callxx : identifikace v Call250V, pořadí činnosti od 0 zvýšení o 1 s předponou A, např. A1, A2, ...

`mon_parametry`(int poboID, string nazev);

Vrací hodnotu nastavení parametru podle názvu parametru v `nazev` dané pobočky `poboID`; pokud `nazev` není použit, metoda vrací všechny parametry požadované pobočky.

URL dotaz: `json.php?method=mon_dej_poradi`

Tělo URL dotazu:

```
{
  "poboID":    <long>,
  "nazev":     <string>
}
```

Dotaz objektem:

```
{
  "methodName": "mon_parametry",
  "params": {
    "poboID":    <long>,
    "nazev":     <string>
  }
}
```

`poboID`: je celočíselná identifikace pobočky,

`nazev`: je nepovinný název parametru,

Odpověď:

```
{
  "return": [
```

```

        {
            "nazev":          <string>,
            "obsah":          <string>
        },
        ...
    ]
}

```

nazev : název parametru,

obsah : obsah parametru, nastavení pobočky,

## 2.5 Ostatní

`ost_dej_pobocka(long firmaID, long poboID);`

Získání informací o pobočkách určité firmy.

Jsou-li uvedeny současně oba parametry, metoda vrací informace o zadané pobočce, v tomto případě může být parametr `firmaID` prázdný;

Je-li prázdný parametr `poboID`, metoda vrací informace o všech pobočkách firmy dané parametrem `firmaID`;

jsou-li uvedeny oba parametry, musí uvedená pobočka patřit uvedené firmě;

jsou-li oba parametry prázdné, metoda vrací v návratu `Fault() ERR_POBO_OR_FIRMA`;

Jsou-li uvedeny současně oba parametry, metoda vrací informace o zadané pobočce, v tomto případě může být parametr `firmaID` prázdný; je-li prázdný parametr `poboID`, metoda vrací informace o všech pobočkách firmy dané parametrem `firmaID`; jsou-li uvedeny oba parametry, musí uvedená pobočka patřit uvedené firmě; jsou-li oba parametry prázdné, metoda vrací chybový stav `ERR_RECORD_NOTINDB`.

URL dotaz: `json.php?method=ost_dej_pobocka`

Tělo URL dotazu:

```

{
    "firmaID":      <long>
    "poboID":      <long>
}

```

Dotaz objektem:

```

{
    "methodName": "ost_dej_pobocka"
    "params": {
        "firmaID":      <long>
        "poboID":      <long>
    }
}

```

`firmaID`: ID firmy, jejíž pobočky chceme zobrazit

`poboID`: pobočka, jejíž data nás zajímají

Odpověď:

```

{
    "return": [

```

```

    {
        "poboID":      <long>,
        "nazev":       <string>,
        "adresa":      <string>,
        "firmaID":     <long>,
        "firma_nazev": <string>,
        "firma_adresa": <string>,
        "tel":         <string>,
        "mobil":       <string>,
        "email":       <string>,
        "www":         <string>,
        "pozn":        <string>
    },
    ...
]
}

```

poboID:	jednoznačný identifikátor pobočky,
nazev:	název pobočky,
adresa:	adresa pobočky,
firmaID:	celočíselná jednoznačná identifikace firmy pobočky,
firma_nazev:	název firmy,
firma_adresa:	adresa firmy,
tel:	telefonní číslo na pobočku,
mobil:	mobilní telefon pobočky,
email:	email pobočky,
www:	URL pobočky,
pozn:	doprovodný text poznámky k pobočce.

```
ost_dej_pocitadlo(int poboID);
```

Získání všech počítačů přístupů stránek pobočky,

URL dotaz: json.php?method=ost\_dej\_pocitadlo

Tělo URL dotazu:

```

{
    "poboID":    <long>
}

```

Dotaz objektem:

```

{
    "methodName": "ost_dej_pocitadlo"
    "params": {
        "poboID":    <long>
    }
}

```

poboID: ID pobočky, na jejíž počítač se ptáme

Odpověď:

```
{
    "return": [
        {
            "nazev":          <string>,
            "pocitadlo_pristupu":  <long>
        },
        ...
    ]
}
```

nazev:                      název počítačidla,  
pocitadlo\_pristupu:        hodnota počítačidla

`ost_dej_info_soap`(string klic);

Získání informací o vývoji protokolu SOAP pro WebCall.

URL dotaz: json.php?method=ost\_dej\_info\_soap

Tělo URL dotazu:

```
{
    "klic":      <string>
}
```

Dotaz objektem:

```
{
    "methodName": "ost_dej_info_soap"
    "params": {
        "klic":      <string>
    }
}
```

klic:              Název hledaného parametru

Odpověď:

```
{
    "return":
    [
        {
            "hodnota":  <string>,
            "dt_datum": <string>
        }
    ]
}
```

Možné chybové stavy:

ERR\_POC\_PARAM  
ERR\_DB\_QUERY

```
ERR_JSON_NO_METHOD
ERR_JSON_PARSE_ERROR
ERR_JSON_WRONG_ARGS_COUNT
```

```
ost_dej_timestamp();
```

metoda vrací aktuální datum a čas serveru ve formátu počtu vteřin od začátku epochy (od 1.1.1970 00:00:00).

URL dotaz: json.php?method=ost\_dej\_info\_soap

Dotaz objektem:

```
{
  "methodName": "ost_dej_timestamp"
  "params": {}
}
```

Odpověď:

```
{
  "return":
  {
    "pocetvterin":    <long>
  }
}
```