

# General applications

## Операторы

### if

Оператор if управляет условным ветвлением.

```
{
  "if": {
    "expression": "${myVar} == '1'",
    "then": [],
    "else": []
  }
}
```

### switch

Операторы switch и case помогают управлять сложными условными операциями и операциями ветвления.

```
{
  "switch": {
    "variable": "${getIvrDigit}",
    "case": {
      "1": [],
      "2": [],
      "3": [],
      "default": []
    }
  }
}
```

## Триггеры

### disconnected

Выполняет набор приложений после завершения основного скрипта. Объявляется единожды в начале скрипта.

```
{
  "trigger": {
    "disconnected": []
  }
}
```

## Функции

### function

Объединить набор приложений в именную функцию. Объявляется в начале скрипта.

```
{
  "function": {
    "name": "myFunction",
    "actions": []
  }
}
```

## execute

Выполнение предварительно объявленной функции по имени.

```
{
  "execute": {
    "name": "myFunction"
  }
}
```

## Переменные

### set

Назначение переменных на канал.

```
{
  "set": {
    "hangup_after_bridge": "true",
    "ivrLang": "en"
  }
}
```

### export

Экспорт переменных на сторону оператора, а также в статистику.

```
{
  "export": [
    "reverted_caller_id_number"
  ],
}
```

### unSet

Разустанавливает переменную.

```
{
  "unSet": [
    "reverted_caller_id_number"
  ]
}
```

## Служебные приложения

### calendar

Приложение позволяет проверить текущее время с рабочим календарем. Значение сохраняется в переменную.

```
{
  "calendar": {
    "extended": false,
    "name": "WorkDay",
    "setVar": "isWorkDay"
  }
}
```

## httpRequest

Выполнение HTTP REST запроса и обработка результата ([get JSON](#))

```
{
  "httpRequest": {
    "url": "https://sales.bpmonline.com/${id}/dataservice/json
/reply/SelectQuery",
    "method": "POST",
    "timeout": 1000,
    "insecureSkipVerify": false,
    "parser": "application/json",
    "responseCode": "http_response_code",
    "headers": {
      "Content-Type": "application/json",
      "Cookie": "${my_cookie}"
    },
    "path": {
      "id": 0
    },
    "data": {
      "Name": "Supervisor",
      "UserID": "Supervisor"
    },
    "exportVariables": {
      "effective_caller_id_name": "callerIdName",
      "owner_caller_id_number": "callerIdOwner"
    }
  }
}
```

## js

Выполнение произвольного JavaScript кода с возвращением результата в переменную

```
{
  "js": {
    "data": "var time = LocalDate(); time.setDate(time.getDate() +
(+${dpd}*1)); return time.getMonth() + '-' + time.getDate() + '-' + time.
getFullYear()",
    "setVar": "myVar"
  }
}
```

## math

Приложение для работы с математическими функциями

```
{
  "math": {
    "data": "${caller_id_array}",
    "setVar": "new_random_caller_id",
    "fn": "random"
  }
}
```

## fn

- `random`: returns a random number from array
- `min` and `max`: can be used to find the lowest or highest value in a list of arguments
- `round`: rounds a number to the nearest integer
- `ceil`: rounds a number up to the nearest integer
- `floor`: rounds a number down to the nearest integer

## JavaScript Math

## string

Приложение для работы со строкой.

```
{
  "string": {
    "data": "${caller_id_number}",
    "fn": "replace",
    "setVar": "reverted_caller_id_number",
    "args": [
      "/^0/",
      "+84"
    ],
  }
}
```

## fn

- `length`: returns the length of a string
- `indexOf` and `lastIndexOf`: returns the index of (the position of) the first or last occurrence of a specified text in a string
- `search`: searches a string for a specified value and returns the position of the match
- `slice`: extracts a part of a string and returns the extracted part in a new string
- `substring`: is similar to `slice`. The difference is that `substring` cannot accept negative indexes.
- `substr`: is similar to `slice`. The difference is that the second parameter specifies the length of the extracted part.
- `replace`: replaces a specified value with another value in a string
- `toUpperCase` or `toLowerCase`: A string is converted to upper case or to lower case
- `charAt`: returns the character at a specified index (position) in a string
- `charCodeAt`: returns the unicode of the character at a specified index in a string
- `split`: A string can be converted to an array with the `split` function
- `reverse`: Reverse the provided string
- `base64`: base64 encoder / decoder
- `MD5`: Computes a digest from a string using MD5 algorithm
- `SHA-256`: Computes a digest from a string using SHA-256 algorithm
- `SHA-512`: Computes a digest from a string using SHA-512 algorithm

## JavaScript String

## markIVR

Позволяет сохранить массив значений в переменную, которая будет доступна в истории

```
{
  "markIVR": {
    "name": "Log",
    "value": " "
  }
}
```

## userInfo

Приложение позволяет получить значение из объекта пользователя по номеру телефона

```
{
  "userInfo": {
    "set": {
      "usrId": "id",
      "mob": "variables.fwd_mob",
      "out": "variables.out_cid"
    },
    "user": {
      "extension": "${ext}"
    }
  }
}
```

## setGrantee

Передать права на канал пользователю с идентификатором указанным в id

```
{
  "setGrantee": {
    "id": 10
  }
}
```

## schema

Выполнить внешнюю схему по идентификатору

```
{
  "schema": {
    "id": 1
  }
}
```

## softSleep

Пауза выполнения скрипта в миллисекундах

```
{
  "softSleep": 1000
}
```

## sql

Приложение доступно исключительно для выделенных либо собственных серверов Webitel, позволяет выполнить запрос в базу данных и вернуть результат в переменную

```
[
  {
    "sql": {
      "dns": "admin:l23Webitel4@tcp(database-1.cluster-cnvuzqnsxyv.
eu-central-1.rds.amazonaws.com:3306)/mysql",
      "driver": "mysql",
      "params": [
        "db"
      ],
      "query": "select l as tst_number, true as tst_bool,
table_name, last_update from mysql.innodb_index_stats where table_name = ?
order by rand() limit 1"
    }
  },
  {
    "sql": {
      "dns": "user=webitel password=webitel host=10.9.8.111
dbname=webitel sslmode=disable",
      "driver": "postgres",
      "params": [
        12
      ],
      "query": "select user_id from call_center.cc_agent where id =
$1",
      "timeout": 5000
    }
  }
]
```

## log

Вывод дополнительной информации в лог файл приложений

```
{
  "log": " " >> ${Hold} <<<"
}
```

## goto

Перейти к именной метке

```
{
  "goto": "tagName"
}
```

## Параметры приложений

Данные приложения могут вызываться как самостоятельно, так и как параметр другого приложения.

## tag

Установить именную метку в скрипте

```
{
  "tag": "tagName"
}
```

## async

Выполнение приложения без блокировки

```
{
  "async": true
}
```

## break

Прерывает выполнение скрипта

```
{
  "break": true
}
```

## limit

Позволяет прерывать цикл после прохождения количества указанного в атрибуте max. Выход осуществляется в именную метку, указанную в атрибуте failover.

```
{
  "limit": {
    "max": 5,
    "failover": "anotherTag"
  }
}
```