

de.intratech.ibs2.servlets

Class PublicIntratechServlet

Object

```

HttpServlet
  ITServlet
    PublicIntratechServlet

```

```

public class PublicIntratechServlet
extends ITServlet

```

Most API endpoints can be called by GET or POST.

Example:

https://ibs.intratech.de/branch_group/PublicIntratechServlet.do?servletAction=exampleAction&...

Method Summary

All Methods	Instance Methods	Concrete Methods
-------------	------------------	------------------

Modifier and Type	Method and Description
BaseResponse	changePasswordByOldPasswordAction (HttpServletRequest request, HttpServletResponse response, BaseResponse resp) Change user password.
BaseResponse	checkCustomerActiveAction (HttpServletRequest request, HttpServletResponse response, BaseResponse responseJson) checks if customer with email and brithday has an active contract
BaseResponse	checkIbanAction (HttpServletRequest request, HttpServletResponse response, BaseResponse resp)
BaseResponse	checkinAction (HttpServletRequest request, HttpServletResponse response, BaseResponse resp)
void	checkTokenValidityAction (PrintWriter out) already checked before, so just a positive answer.
BaseResponse	createAuthTokenAction (HttpServletRequest request, HttpServletResponse response, BaseResponse resp) Create AuthToken for user with password or for given customer
BaseResponse	deleteAuthTokenAction (HttpServletRequest request, HttpServletResponse response, BaseResponse resp) Delete AuthToken given in this request or all AuthToken of user if authorized by password without token
BaseResponse	fileUploadAction (HttpServletRequest request, HttpServletResponse response) file upload POST request with content type "multipart/form-data" max 5MByte per request
BaseResponse	getOpenReceivablesAction (HttpServletRequest request, HttpServletResponse response, BaseResponse resp) Returns a List of open Receivables.
BaseResponse	initISalesAction (HttpServletRequest request, HttpServletResponse response, BaseResponse resp)
void	loadBaseDataAction (HttpServletRequest request, HttpServletResponse response, PrintWriter out) Returns the base data of the customer.
void	loadBranchListAction (HttpServletRequest request, HttpServletResponse response, PrintWriter out) Returns a list with all branches the user has access to.
void	loadCalendarDatesConfigAction (HttpServletRequest request, HttpServletResponse response, PrintWriter out) Loads selfservice configuration of employee * branches * activities * courses
void	loadContractDataAction (HttpServletRequest request, HttpServletResponse response,

	<pre>PrintWriter out) Loads active contract data of logged in customer.</pre>
BaseResponse	<pre>loadNumberOfPresentCustomers(HttpServletRequest request, HttpServletRequest response, BaseResponse resp) Load number of actual checked in customers and configuration of maxCheckinsAllowed</pre>
BaseResponse	<pre>loadTimestopAction(HttpServletRequest request, HttpServletResponse response, BaseResponse resp) Load timestops of customer</pre>
void	<pre>loginAction(HttpServletRequest request, HttpServletResponse response, PrintWriter out) Gives some basic personal data for eg greeting.</pre>
void	<pre>reservationDate(HttpServletRequest request, HttpServletResponse response, PrintWriter out)</pre>
BaseResponse	<pre>searchCustomerAction(HttpServletRequest request, HttpServletResponse response, BaseResponse responseJson) Search customer by customerId,name,birthday,email.</pre>
BaseResponse	<pre>sendEmailAction(HttpServletRequest request, HttpServletResponse response, BaseResponse resp)</pre>
BaseResponse	<pre>storeCommentAction(HttpServletRequest request, HttpServletResponse response, BaseResponse resp) Store a comment for a customer</pre>
BaseResponse	<pre>storeContractAction(HttpServletRequest request, HttpServletResponse response, BaseResponse responseJson)</pre>
BaseResponse	<pre>storeCustomerAction(HttpServletRequest request, HttpServletResponse response, BaseResponse responseJson) Store new customer or update customer information.</pre>
BaseResponse	<pre>storeCustomerCampaignAction(HttpServletRequest request, HttpServletResponse response, BaseResponse resp)</pre>
BaseResponse	<pre>storeCustomerGroupsAction(HttpServletRequest request, HttpServletResponse response, BaseResponse resp)</pre>

Method Detail

changePasswordByOldPasswordAction

```
public BaseResponse changePasswordByOldPasswordAction(HttpServletRequest request,
                                                    HttpServletResponse response,
                                                    BaseResponse resp)
```

Description copied from class: **PublicIntratechServlet**

Change user password.

Authorization parameter:

user - String

password - String - for use with user

authToken - String - as alternative to user and password

Mandatory parameter:

newpassword - String - The password to be set.

checkCustomerActiveAction

```
public BaseResponse checkCustomerActiveAction(HttpServletRequest request,
                                                    HttpServletResponse response,
                                                    BaseResponse responseJson)
```

Description copied from class: **PublicIntratechServlet**

checks if customer with email and brithday has an active contract

Authorization parameter:

user - String

password - String - for use with user

authToken - String - as alternative to user and password

Optional parameter:

birthday - String - format dd.MM.yyyy

email - String - customer email

Result example:

```
{"authentication":true,"servletAction":"checkCustomerActiveAction","status":true,"message":"1 aktive Kunden gefunden.", "count":1}
```

checkIbanAction

```
public BaseResponse checkIbanAction(HttpServletRequest request,
                                    HttpServletResponse response,
                                    BaseResponse resp)
```

Authorization parameter:

user - String

password - String - for use with user

authToken - String - as alternative to user and password

Mandatory parameter:

iban - String - the IBAN to check

Result example:

```
{"authentication":true,"servletAction":"checkIbanAction","status":true,"bankData":
{"bic":"BYLADEM1001","id":20745,"name":"Deutsche Kreditbank Berlin"}}
```

checkTokenValidityAction

```
public void checkTokenValidityAction(PrintWriter out)
```

Description copied from class: [PublicIntratechServlet](#)

already checked before, so just a positive answer.

Authorization parameter:

authToken - String - to be checked

checkinAction

```
public BaseResponse checkinAction(HttpServletRequest request,
                                   HttpServletResponse response,
                                   BaseResponse resp)
```

Authorization parameter:

authToken - String

Mandatory parameter:

branchId - Integer

createAuthTokenAction

```
public BaseResponse createAuthTokenAction(HttpServletRequest request,
                                          HttpServletResponse response,
                                          BaseResponse resp)
```

Description copied from class: [PublicIntratechServlet](#)

Create AuthToken for user with password or for given customer

Authorization parameter:

user - String

password - String - for use with user

authToken - String - as alternative to user and password

Optional parameter:

birthday - Date - dd.MM.yyyy

customerId - Integer

email - String - Can be used as loginname if no user exists

deleteAuthTokenAction

```
public BaseResponse deleteAuthTokenAction(HttpServletRequest request,
                                         HttpServletResponse response,
                                         BaseResponse resp)
```

Description copied from class: PublicIntratechServlet

Delete AuthToken given in this request or all AuthToken of user if authorized by password without token

Authorization parameter:

user - String

password - String - for use with user

authToken - String - as alternative to user and password

Optional parameter:

authToken - String - delete this token

fileUploadAction

```
public BaseResponse fileUploadAction(HttpServletRequest request,
                                     HttpServletResponse response)
```

Description copied from class: PublicIntratechServlet

file upload POST request with content type "multipart/form-data" max 5MByte per request

Authorization parameter:

user - String

password - String - for use with user

Mandatory parameter:

file - file - to be uploaded

customerId - int - assign to customer

fileType - int - 1:document 2:photo

Result example:

```
{"authentication":true, "status":false, "message":"file size exceeded."}
```

getOpenReceivablesAction

```
public BaseResponse getOpenReceivablesAction(HttpServletRequest request,
                                             HttpServletResponse response,
                                             BaseResponse resp)
```

Description copied from class: PublicIntratechServlet

Returns a List of open Receivables.

Authorization parameter:

user - String

password - String - for use with user

Mandatory parameter:

contractId - int - id of contract

initISalesAction

```
public BaseResponse initISalesAction(HttpServletRequest request,
                                     HttpServletResponse response,
                                     BaseResponse resp)
```

Authorization parameter:

user - String

password - String - for use with user

authToken - String - as alternative to user and password

Optional parameter:

branchId - int - loads contract templates and systemgroups

Result example:

```
{ "authentication": true, "servletAction": "initISalesAction", "contractTemplateArray": [{ "code": "",
"description": "Super Vertrag", "amount": 0.0, "duration": 1, "durationUnit": 1, "accountingCycle": 1,
"accountingCycleUnit": 1, "prePaid": false, "prePaidPercent": 0, "contractTemplateModulArray": [{ "id": 2,
"name": "Getränke", "description": "Getränke", "price": 0.0 }, { "id": 1, "name": "Check-In", "description":
"Check-In", "price": 0.0 }], "extraReceivableArray": [], "id": 179, "name": "Aktion XYZ" }],
"systemGroupsArray": [{ "multiallocation": false, "systemGroupsSubArray": [{ "multiallocation": false, "id":
3, "name": "Werbefläche" }, { "multiallocation": false, "id": 4, "name": "mündliche Empfehlung" }, {
"multiallocation": false, "id": 5, "name": "Internet" }], "id": 2, "name": "Werbequellen" } ] }
```

loadBaseDataAction

```
public void loadBaseDataAction(HttpServletRequest request,
                             HttpServletResponse response,
                             PrintWriter out)
```

Description copied from class: **PublicIntratechServlet**

Returns the base data of the customer.

Authorization parameter:

user - String

password - String - for use with user

authToken - String - as alternative to user and password

loadBranchListAction

```
public void loadBranchListAction(HttpServletRequest request,
                                HttpServletResponse response,
                                PrintWriter out)
```

Description copied from class: **PublicIntratechServlet**

Returns a list with all branches the user has access to.

Authorization parameter:

user - String

password - String - for use with user

authToken - String - as alternative to user and password

Optional parameter:

branchId - int - branchId of requested branch

loadCalendarDatesConfigAction

```
public void loadCalendarDatesConfigAction(HttpServletRequest request,
                                           HttpServletResponse response,
                                           PrintWriter out)
```

Description copied from class: **PublicIntratechServlet**

Loads selfservice configuration of employee * branches * activities * courses

Authorization parameter:

user - String

password - String - for use with user

authToken - String - as alternative to user and password

loadContractDataAction

```
public void loadContractDataAction(HttpServletRequest request,
                                   HttpServletResponse response,
                                   PrintWriter out)
```

Description copied from class: **PublicIntratechServlet**

Loads active contract data of logged in customer. Returns contracts, contractProperties, open receivables and timeoffs.

Authorization parameter:

user - String

password - String - for use with user

authToken - String - as alternative to user and password

loadNumberOfPresentCustomers

```
public BaseResponse loadNumberOfPresentCustomers(HttpServletRequest request,
                                                HttpServletResponse response,
                                                BaseResponse resp)
```

Description copied from class: [PublicIntratechServlet](#)

Load number of actual checked in customers and configuration of maxCheckinsAllowed

Authorization parameter:

user - String

password - String - for use with user

Mandatory parameter:

branchId - int - id of branch

Result example:

```
{"authentication":true,"servletAction":"loadNumberOfPresentCustomers","status":true,"checkinData":
{"maxCheckinsAllowed":10,"countCheckedInCustomer":2}}
```

loadTimestopAction

```
public BaseResponse loadTimestopAction(HttpServletRequest request,
                                       HttpServletResponse response,
                                       BaseResponse resp)
```

Description copied from class: [PublicIntratechServlet](#)

Load timestops of customer

Authorization parameter:

user - String

password - String - for use with user

Mandatory parameter:

customerId - int - id of branch

Result example:

```
{"authentication":true,"servletAction":"loadTimestopAction","status":true,"timestops":[]}
```

loginAction

```
public void loginAction(HttpServletRequest request,
                        HttpServletResponse response,
                        PrintWriter out)
```

Description copied from class: [PublicIntratechServlet](#)

Gives some basic personal data for eg greeting.

Authorization parameter:

user - String

password - String - for use with user

authToken - String - as alternative to user and password

Result example:

```
{"authentication" : true, "user": {"firstname" : "Hans", "lastname" : "Meier", "employee" : true, "branchId"
: 9, "branchName" : "Demo Studio", "customerId" : 2356, "sex":0}}
```

reservationDate

```
public void reservationDate(HttpServletRequest request,
                             HttpServletResponse response,
                             PrintWriter out)
```

Mandatory parameter:

authToken

searchCustomerAction

```
public BaseResponse searchCustomerAction(HttpServletRequest request,
                                         HttpServletResponse response,
                                         BaseResponse responseJson)
```

Description copied from class: PublicIntratechServlet

Search customer by customerId,name,birthday,email.

Authorization parameter:

user - String

password - String - for use with user

authToken - String - as alternative to user and password

Optional parameter:

firstname - String - allows wildcard * in String

lastname - String - allows wildcard * in String

email - String

branchId - int

customerId - int

Action result:

ResponseData with customerList (max 100 entries)

Result example:

```
{"authentication":true,"servletAction":"searchCustomerAction","status":true,"customerList":
[{"customerId":2787,"firstname":"Testa","lastname":"Burger","birthday":"11.10.1978"}]}
```

sendEmailAction

```
public BaseResponse sendEmailAction(HttpServletRequest request,
                                     HttpServletResponse response,
                                     BaseResponse resp)
```

Authorization parameter:

user - String

password - String - for use with user

authToken - String - as alternative to user and password

Mandatory parameter:

customerId - int - The customers Id

contractId - int - Send contract pdf of this contract

Optional parameter:

email - String - Overrule persons email

storeCommentAction

```
public BaseResponse storeCommentAction(HttpServletRequest request,
                                       HttpServletResponse response,
                                       BaseResponse resp)
```

Description copied from class: PublicIntratechServlet

Store a comment for a customer

Authorization parameter:

user - String

password - String - for use with user

authToken - String - as alternative to user and password

Mandatory parameter:

customerId - int - The customers Id

text - String - The comment text

Optional parameter:

title - String - The comment title

type - [1,2,3] - Type of comment. defaults to 1 (PopUp)

storeContractAction

```
public BaseResponse storeContractAction(HttpServletRequest request,
                                       HttpServletResponse response,
                                       BaseResponse responseJson)
```

Authorization parameter:

user - String

password - String - for use with user

authToken - String - as alternative to user and password

Mandatory parameter:

customerId - int

contractTemplateId - int - Create contract of this template

startContract - Date - Format dd.MM.yyyy

startTraining - Date - Format dd.MM.yyyy

idPaymentType - int - refers to

Optional parameter:

accountingDay - int - day of month accounting should proceed

accountingGroupId - int - accountingGroupId

branchId - int

prePaid

prePaidPercent - int

preUseAmount - String

comment - String

advanceuse - String - Create receivable for usage before contractBegin

extraReceivableArray - JSONArray - [{"id":1,"amount":0,01},...]

contractModulArray - JSONArray - [{"id:1"},...]

promotionCustomerId - boolean

Result example:

```
{"authentication":true,"servletAction":"storeContractAction","status":true,"contractId":255025}
```

storeCustomerAction

```
public BaseResponse storeCustomerAction(HttpServletRequest request,
                                       HttpServletResponse response,
                                       BaseResponse responseJson)
```

Description copied from class: PublicIntratechServlet

Store new customer or update customer information.

Authorization parameter:

user - String

password - String - for use with user

authToken - String - as alternative to user and password

Mandatory parameter:

firstname - String

lastname - String

birthday - String - Format dd.MM.yyyy

street - String

city - String

postcode - String

sex - [0,1] - 0:male 1:female

Optional parameter:

id - int - customerId for modifying

branchId - int

countryId - int - id of country defaults to country of branch on new customer

languageId - int - id of language defaults to country of branch on new customer

phone - String

email - String
mobil - String
calling - boolean
callingWork - boolean
callingMobil - boolean
smsNews - boolean
emailNews - boolean
letterNews - boolean
noCalling - boolean
bicBank - String
blzBank - String
accountHolderBank - String
ibanBank - String
accountnumber - String
appUsage - boolean
useType - [customer,lead] - for mandatory field selection. defaults to customer

Result example:

```
{"authentication":true,"servletAction":"storeCustomerAction","status":true,"customerId":278727}
```

storeCustomerCampaignAction

```
public BaseResponse storeCustomerCampaignAction(HttpServletRequest request,  
                                               HttpServletResponse response,  
                                               BaseResponse resp)
```

Authorization parameter:

user - String
password - String - for use with user
authToken - String - as alternative to user and password

Mandatory parameter:

customerId - Integer

Optional parameter:

campaignId - Integer - campaign to be stored. if called without parameter or zero campaign of customer will be deleted
costsId - Integer
systemGroupsArrayId - String - JSON encoded array with objects

Result example:

```
{"authentication" : true, "customerId": 12, "status": true}
```

storeCustomerGroupsAction

```
public BaseResponse storeCustomerGroupsAction(HttpServletRequest request,  
                                              HttpServletResponse response,  
                                              BaseResponse resp)
```

Authorization parameter:

user - String
password - String - for use with user
authToken - String - as alternative to user and password

Mandatory parameter:

customerId - Integer
systemGroupsArrayId - String - JSON encoded array with objects
[{"id":"93","multiallocation":true,"systemGroupsSubArray":[{"id":98},{"id":101}],
{"id":"106","multiallocation":true,"systemGroupsSubArray":[{"id":114},{"id":115}]]

Result example:

```
{"authentication" : true, "customerId": 12, "status": true}
```

PACKAGE **CLASS** DEPRECATED INDEX HELP

PREV CLASS **NEXT CLASS** FRAMES NO FRAMES ALL CLASSES

SUMMARY: NESTED | FIELD | CONSTR | METHOD DETAIL: FIELD | CONSTR | METHOD