

# Network Protocol

## 1 Overview

Below are some example commands used in the game protocol. Note that all commands include a concealed parameters section appended after the visible parameters. This section always follows the structure:

Furthermore, all parameters use type prefixes:

- **S:** for String
- **I:** for Integer
- **F:** for Float
- **D:** for Double
- **B:** for Boolean
- **L:** for Long

## 2 Commands

### 2.1 Create a New Game Object

Command:

```
CREATE {REQUEST} [S:Characters, S:USERNAME, F:positionX, F:positionY, F:width, S:
    Gamesession]
```

This command creates a new game object with the specified parameters.

### 2.2 Create a Specific Object

For example, for NinjaGuy and Ricardo using BandageGuy:

```
CREATE {REQUEST} [S:Ricardo,S:USERNAME,S:GAMESESSION,F:40.0,F:40.0,S:src/main/java/ch/
    unibas/dmi/dbis/cs108/example/ClientServerStuff/resources/ricardo.png]
```

This command creates a Ricardo object with the provided parameters.

### 2.3 Login Command

```
LOGIN {REQUEST} [S:USERNAME]
```

This command connects the client's UUID to the GUI and assigns keybinds.

### 2.4 Exit Command

```
EXIT {REQUEST} []
```

This command terminates the client.

## 2.5 Delete Command

```
DELETE {REQUEST} [S:USERNAME]
```

This command deletes the specified player.

## 2.6 Logout Command

```
LOGOUT {REQUEST} [S:USERNAME]
```

This command handles logout of the specified player.

## 2.7 Chat Command

```
CHAT {REQUEST} [S: hello]
```

This command sends a chat message. The concealed parameters follow the same structure as noted above.

## 2.8 change name

```
CHANGENAME{REQUEST} [S:USERNAME,S:NEWUSERNAME]
```

This command changes the name of the specified player.

## 2.9 Fast Login

```
FAST_LOGIN{CLIENT} []
```

This command will call CREATE and LOGIN simultaneously.

## 2.10 Change Username

```
CHANGE_USERNAME{CLIENT} [S:USERNAME]
```

This command changes the local username of the client.

### 3 Generalized Command Format

The general format for commands in this protocol is:

`<COMMAND> {OPTION} [<Parameter List>] | <Concealed Parameters> |`

where:

- `<COMMAND>` is the command type (e.g., CREATE, LOGIN, EXIT, etc.).
- `<OPTION>` can be:
  - **REQUEST:** for commands,
  - **RESPONSE:** for server answers,
  - **GAME:** for non-checked messages.
  - **CLIENT:** for the change of Client local state.
- `<Parameter List>` is a comma-separated list of parameters using type prefixes (e.g., S: for String, F: for Float).
- `<Concealed Parameters>` is always a fixed sequence appended after a pipe delimiter:

`S: gameObjectName, S: GameSessionName, S: username, S: messageUUID`