

# Network protocol

This document lists all network protocol commands that are used for the project and will be completed as we go.

**Basic structure of command:** *COMMAND\$Option\$Argument1\$Argument2...*

Command is always an all-caps 5-letter word, while option allows the program to use the same command for similar actions. While there is only one option per command until now, further ones will be added.

**Example:** *PRINT\$Terminal\$Welcome to Skip-Bo*

*This command asks the receiving client to print „Welcome to Skip-Bo“ on its terminal.*

Section	Command	Option	Arguments	Description
General	PRINT	Terminal	String message	prints ,message' on receiver's terminal
Lobby	SETTO	Nickname	String nickname	sets name of new player to ,nickname'
	CHNGE	Nickname	String nickname	changes name of sending player to ,nickname'
		Status	PlayerStatus ps	sets senders PlayerStatus to ,ps'
	LGOUT	-	-	from client: logout request to server, from server: logout confirmation to client
	DISPL	players	-	gets a list of all players currently connected to the server
		games	-	gets a list of all ongoing and finished games
	PLAYR	List	String nicknames	Tells client which players are already connected to the server
		Joined	String nickname	Tells client that the player with the name "nickname" joined the server

Section	Command	Option	Arguments	Description
Chat		Left	String nickname	Tells client that the player with the name "nickname" left the server
		Change	String oldName, String newName	Tells client that the player with the name "oldName" change its name to "NewName"
	CHATM	Global	String message	sends ,message' to write on the global chat (,global' chat being either the lobby or the game the player is in)
		Private	Player p, String message	sends private ,message' to player p
		Broadcast	String message	sends ,message' to all players, regardless of both the sender's and the receivers' locations
Game	NWGME	New	int n, int x	starts new game with n players and a stockpile that consisting of x cards
		Names	int x, Nx {String name}	confirms to client that game with x stockcards is starting, with the names of the N players in that game as argument
		Cards	5x {Colour c, int number}, Nx {String name. colour c, int number}	tells client which cards the player has at the beginning of the game (first 5 are hand cards, then the top stockpile cards of the N players in the game)
	ENDGM	Winner	String name	signals client that game is ending, player with ,name' won
		Terminated	-	signals client that game ended without a winner because every player except one left
	PUTTO	Card	Pile from, int i, Pile to, int j	requests to put card from pile ,from', index ,i', to pile ,to', index ,j'

Section	Command	Option	Arguments	Description
		Response	Pile from, int i, Pile to, int j, String ,name', Colour c, int x	tells client to execute the move (as stated above, card has colour ,c' and number ,x') by player with name ,name'
		StockResponse	Pile from, int i, Pile to, int j, Player p, Colour c1, int x1, Colour c2, int x2	tells client to execute the move (as stated above) by player ,p' plus what the new top card of the stockpile is (colour ,c1' and number ,x1') and which card has been moved (colour ,c2' and number ,x2')
	CHECK	HandCards	5x {Colour c, int number}	tells client which hand cards the player has
		StockCard	Colour c, int number	tells client what its top card of the stockpile is