Software Engineering Assignment 1

Zubair Abid

On initialising the `Args` object, with a schema and an argument list, we: - Create: - `marshalers`, a map from characters to marshalers - `argsFound`, a set of all characters found so far - Parse: - the schema, using `parseSchema`: - the schema (which is a comma-separated list of `char+indicator` combinations) is split by commas, and: - for each *comma-separated element* of non-zero length, we pass it to `parseSchemaElement` where each element is split into: - an `elementId` - which has to be a single character to be valid (the validator, `validSchemaElementId` just checks if the character is a letter or not), - and the rest of the string, which has to be a specific character combination. This combination is used to decide which `ArgumentMarshaler` will be associated with the `elementId` - So, at the end of this you have a new entry to the `marshalers` map: the `elementId` and the associated marshaler, both extracted from `element` - Now, **`marshalers` is a map for all comma-separated elements of the schema** - the argument list, using `parseArgumentStrings`: - going over the list, it passes all elements starting with '-' to `parseArgumentCharacters`: - a wrapper to send each character into `parseArgumentCharacter`, where: - an `ArgumentMarshaler` is gotten from the `marshalers` map for the character.

Schema

Args

Marshalers