Value <<enum>> **Type** writeTo(out:std::ostream &): void +Integer t +clone(): Value * +Float t +type(): Type t +Boolean t +dumps(): std::string +String t +operator <<(out:std::ostream &): std::ostream & +List t +save(filename:const char*): void +Dict t +save(filename:std::string const &): void T:typename Number List **Dict** content: std::vector<Value*> content: std::map<std::string, Value*> - value: T - writeTo(out:std::ostream &): void - writeTo(out:std::ostream &): void - writeTo(out:std::ostream &): void +<<getter>> type(): Type +<<getter>> value(): T +<<getter>> type(): Type +clone(): Value* +clone(): Value* +<<getter>> doubleVal(): double +hasKey(key:std::string const &): bool +steal(index:size t): Value* +<<getter>> intVal(): int +append(val:Value*): void +<<getter>> get(key:std::string const &): Value* +clone(): Value * +<<conversion>> operator T(): T const & +appendPtr(val:Value*): void +steal(key:std::string const &): Value* +<<getter>> len(): size t +set(key:std::string const &,val:Value*): void +<<getter>> operator[](index:size t): Value * +setPtr(key:std::string const &,val:Value*): void +<<getter>> len(): size t +stealMerge(other:Dict &): void **Float** Integer +type(): Type +type(): Type String - content: std::string -_writeTo(out:std::ostream &): void Bool +<<getter>> type(): Type +clone(): Value* - writeTo(out:std::ostream &): void +<<getter>> value(): std::string const & +type(): Type +<<conversion>> operator std::string const &(): std::string const & +<<conversion>> operator const char *(): const char *