Autohaus – Constantin Schultz Assoziation Vererbung Erstellung Autohaus, GUI login LoginFrame Autohaus.autohaus.Autohaus Aggregation Autohaus.user.User credentialManage Settings Autohaus. GUI.manager. GUI parent displayname password_entry : Entry Komposition credentialManager usemame username entry: Entry autohaus known models : dict create_widgets() submit() currentFrame name: str frames : list user : NoneType change frame(frame) vehicle enum: int Autohaus.GUI.standardView.StandardView load protocol() vehicles: list[vehicles.Vehicle] login() add vehicle(vehicle) on closing() standardView() build part(data) create_velucle() create_widgets() close() get brands() get models(brand) Autohaus. GUI. creator. Creator get vehicles() Settings Autohaus vehicles vehicle Vehicle autohaus load config() autohaus logout : Button info : Label brand entry engine engine frame parent vsb : Scrollbar description load vehicles() parent user : Label brand_label : Label gearbox gearbox_frame parent submit_button : Button image_path model chassis entry: Combobox create_widgets() logout() create_widgets() onFrameConfigure(event logout() chassis label : Label storage_path : NoneType vehicle_enum : NoneType save dynamic config() color_entry : Entry tire frame color label : Label vehicle : NoneType set user(user) engine entry: Combobox create_required_widgets() create_widgets() load_protocol() engine_label : Label gearbox entry: Combobox Autohaus. GUI. vehiclelist. VehicleLabel on_closing() submit() gearbox label : Label credentialManager Settings image button : Button vehicle image label : Label view button : Buttor image path : NoneType, tuple Autohaus. GUI. constructionwindows.partframe.PartFrame Autohaus.settings.Settings model entry entries : list parent create widgets() model label : Label view() Autohaus.credentialManager.CredentialManager BASE DIR parent required parts parts : dict storage_path price_entry : Entry check_if_filled() create_required_widgets() get_part() get_values() CARS DIR price_label : Label credentials : dict submit button : Button IMAGE DIR tire entry: Combobox check username() MOTORCYCLES DIR tire label : Label type_entry : Combobox STATIC DIR get user() Antohaus. GUL constructionwindows.motorcycle.MotorcycleConstructionWindow type_label : Label validate login() VEHICLES DIR chassis frame data engine engine_frame gearbox gearbox_frame create image upload() Autohaus. GUI vehiclelist. VehiclePopup create required widgets() image : PhotoImage create_widgets() image_label : Label load_protocol() on_closing() parent submit_button : Button create_widgets() process_image(original_path, new_path) load protocol() Autohaus. GUI. autocompleteentry. AutocompleteEntry Part autocomplete func entry : Entry listbox : Listbox Tire check key(event) required_parts parts : dict storage_path Chassis Engine check_if_filled() create_required_widgets() get_part() get_values() select value(event) consumption milage size milage update listbox() power calculate condition() get_required_values() CombustionEngine WinterTire LuxuryChassis SportsChassis ElectricEngine SummerTire AutomaticGearbox SportsTire fuel_type displayname comfort airfriction charge cycles displayname : str infotext grip infotext spikes calculate efficiency/ calculate comfort() calculate friction() calculate battery condition() calculate_drip() get_required_values() calculate_emissions() get_required_values() get_required_values() get_required_values() get required values()