

Endpoint	Path	ID	Summary	Label	Description	Importance Reason	Input Data	Expected Result	Automated?
/people	/schema	TC_P_1	Validate structure	positive case	Check that there is schema for people resource	We need to be able to get schema as a source of truth for api resources structure, esp. structure of Character entity.	---	JSON schema describing People resource.	yes
	/	TC_P_2		positive case	Check that all swapi characters are retrieved if no additional path data is provided (people count=87).	There should be a way to get all the characters, since we still don't know their properties to filter by.	---	Response with own properties and list of people entities.	yes
		TC_P_3		positive case	Validate endpoint response structure.	In order to manipulate with response correctly we need to make sure we know its structure.	---	Response structure is an object and has the following properties: 'count', 'next', 'previous', 'results'.	yes
		TC_P_4		positive case	Validate "results" array of people objects.	We need to be able to find characters in the response.	---	People is in the "results" array of the response object.	yes
		TC_P_5		positive case	Validate character object structure (all people properties are required).	In order to check character's property we need to validate its entity structure by schema.	---	Character entity structer follows the schema.	yes
	/id	TC_P_6	Check entity	positive case	Validate that each character has films URLs that this character has been in.	Logically People are participators of at least one film, so we should validate that.	---	Array of films exists in each character entity.	yes
		TC_P_7		positive case	Check that the film object is retrieved if following a film URL listed in the "films" property of a character.	We need to make sure URLs to the films work and we can retrieve film data be each URL.	---	Film objects with specific properties such as title.	yes
		TC_P_8		positive case	Check that URL to the character object is listed in the "characters" property of a film.	If a character has a film mentioned in its list of films, so film must have this character listed in the cast as well.	---	Character object with specific properties such as name.	yes
	/id	TC_P_9	Check filtration	positive case	Check it is possible to get a character from people by providing its ID in the path .	People resource supports filtering the people by character ID. Does it? :)	ID = 1	A 'Luke Skywalker'	yes
		TC_P_10		negative case	Check that 404 is returned if out-of-range ID is provided (less than the first ID).	We can get the first and the last character ID by getting and parsing the people entities. So we have boundary values to be tested. Since character ID is a part of URL path its unexisting value in the routing results in "page not found".	ID = 0	404 error	yes
	/?search=<name>	TC_P_11		positive case	Check it is possible to get a character by its name (exact match).	API supports searching by resource. We need to check that we can use it in case we know exactly whom we are searching for.	C-3PO	C-3PO	yes
		TC_P_12		positive case	Check it is possible to get a character by its name (single partial match).	We should be able to get a character if it only partially matches the criteria.	---	---	no
		TC_P_13		positive case	Check it is possible to get some characters by name (multiple partial match).	We may predict that multiple match may occur. We should check that we can get all the matched characters.	search='Lu'	Luke Skywalker', 'Luminara Unduli'	yes
		TC_P_14		negative case	Check that searching by people results in all existing people if search criteria is an empty value.	The API should not fail if we forgot to provided search value.	search=""	total objects count=87, first page of objects retrieved (next page is 2), first batch of 10 people in the "results".	yes
		TC_P_15		positive case	Check that searching by people results in an empty array if search criteria does not match any of existing names.	We should be able to get an empty response if no match was found.	search='qwerty'	count=0, no people objects in the "results".	yes
	/?page=<id>	TC_P_16	Check pagination	positive case	Check that the first page is retrieved if no page id is provided in the URL.	People resource responses with objects per pages. The default qyeru to people endpoint should result in retrieving of the first page always.	---	"next" page URL = 'https://swapi.co/api/people/?page=2', first batch of 10 people in the "results".	yes

		TC_P_17	positive case	Check that the page is returned if its page ID is provided in the URL (page count = 9).	We can get how many pages of people there are. And we should be able to get any of existing pages explicitly by providing its number in the URL.	page=5	"next" page URL = https://swapi.co/api/people/?page=6 , batch of 10 people in the "results".	yes
		TC_P_18	positive case	Check that "next" page URL has not NULL value on the first page of people result.	Logically being on the very first page of the range, the next page should exist and its URL should be defined.	---	---	no
		TC_P_19	positive case	Check that the next page is reached if following the "next" page URL.	Defined next page URL in the response object property should be valid and lead us to that page.	---	---	no
		TC_P_20	positive case	Check that "previous" page URL has not NULL value on the last page of people result.	Logically previous page exists if we are on the last page.	https://swapi.co/api/people/?page=9	"previous" property is not empty.	yes
		TC_P_21	positive case	Check that the previous page is reached if following the "previous" page URL.	Previous page URL should be valid and lead us to that page.	https://swapi.co/api/people/?page=8	"next" property has value = 'https://swapi.co/api/people/?page=9'	yes
		TC_P_22	positive case	Check that "next" page URL is NULL on the last page of people result.	There is no next pages if we are already on the last one.	---	---	no
		TC_P_23	positive case	Check that "previous" page URL is NULL on the first page of people result.	There is no previous pages if we are on the first one.	---	---	no
		TC_P_24	negative case	Check that 404 error is returned if invalid page ID is provided in the URL.	All the pages have its content and addresses to them are defined in the app routing, if such page address is not there - it cannot be found.	---	---	no
		TC_P_25	negative case	Check that 404 error is returned if negative page ID is provided (-1).	Minus must be counted and this index must not behave as page number 1, unless there is such feature :).	---	---	no