Places

#	Endpoint	Purpose	Input Data	Excepted output	Actual Output	Result
1	POST /api/v1/places/	Validate creation of a place	{ "user_id":self.owner_id, "title":"Cozy Flat", "description":"Nice 2-bedroom flat near city center", "price":240.0, "address":"888 Main St, Melbourne", "latitude":-24.7241, "longitude":124.6281, "image_url":"https://example.com/image.jpg", "amenity_ids":["1", "2", "2", "3"] }	Status: 201; Response echoes fields; amenity_ids deduped to ["1","2","3"]	Status: 201; Fields match; amenity_ids = ["1","2","3"]	✅ Passed
2	POST /api/v1/places/	Validate creation fails with invalid user	{ "user_id": "invalid-user", "title": "Invalid Place", "price": 240, "latitude": -24.7241, "longitude": 124.6281 }	Status: 400	Status: 400	✅ Passed
3	GET /api/v1/places/{place_id}	Retrieve a single place by ID	Valid place id from row 1	Status: 200; Body contains the same id	Status: 200; id matches	✅ Passed
4	GET /api/v1/places/{InvalidID}	Invalid ID should 404	-	Status: 404	Status: 404	✅ Passed
5	GET /api/v1/places/	Retrieve all places	-	Status: 200; JSON array; contains place_id from row #1	Status: 200; Array includes created id	✅ Passed
6	PUT /api/v1/places/{place_id}	Update a place	Body: { "title":"Updated Flat", "price":200 }	Status: 200; Body reflects updates	Status: 200; Title updated	✅ Passed
7	PUT /api/v1/places/{InvalidID}	Update with invalid ID	Body: { "title":"Fail Update" }	Status: 404	Status: 404	✅ Passed