**Title:** Redis HDEL

***Excerpt:*** Redis hash is an array of string field-value pairs. The HDEL command is used to remove one or multiple fields from a Redis hash stored at a given key. This command returns an integer reply which indicates the count of removed fields.

**Permalink:** redis-hdel

**Category:** Redis

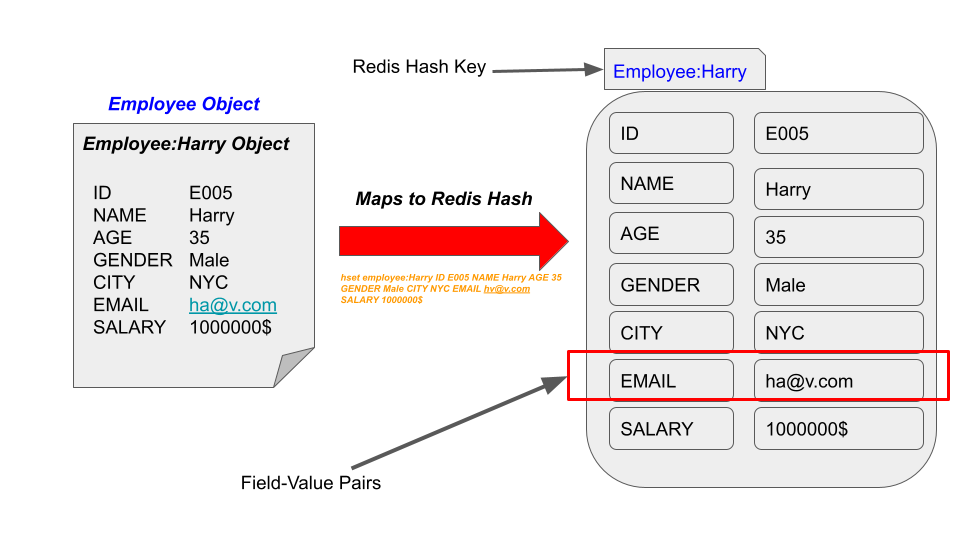
# 

# 

# 

# Redis Hash Fields

Redis hashes are a collection of string-type field-value pairs. Hence, it is an ideal data structure to hold objects with many properties. The great thing about Redis hashes is they can store more than four billion field-value pairs in a small amount of memory which supports you to hold hundreds of objects in very little space.



As shown in the above illustration, we can easily map a real-world entity to a Redis hash. The HSET command is used to create a new hash and add field-value pairs to it.

# The HDEL Command

The HDEL command is used to remove one or more field value pairs from a Redis hash stored at a given key. Since this command accepts multiple field arguments, the time complexity is directly proportional to the number of fields to be removed.

## Syntax:

| HDEL hash\_key field [field ...] |
| --- |

***hash\_key*** : The Redis key of the hash.

***field*** : The name of the field that you want to remove.

This command gives an integer reply which is the number of fields removed from the specified hash. If you have specified non-existing fields, those will be ignored and not counted.

In addition, whenever the key of the hash doesn’t exist, the HDEL command will return 0.

## Use Case 01 - Remove Unwanted User Details

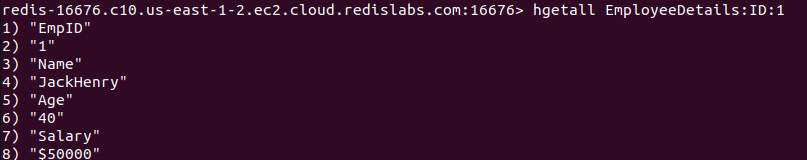
Let’s assume a scenario where we want to store thousands of employee details of a company. Since Redis hashes are more memory-optimized, it is the ideal candidate for this kind of work.

First, we are going to create a new hash *EmployeeDetails:ID:1* to store the details of an employee.The HSET command will be used.

| hset EmployeeDetails:ID:1 EmpID 1 Name JackHenry Age 40 Salary $50000 |
| --- |

Next, we will verify that the hash has been added successfully. The HGETALL command can be used to query details related to a specified hash key.

| hgetall EmployeeDetails:ID:1 |
| --- |



Let’s say the company management thought about outsourcing the payroll processing to an external party. Hence, the salary field in the hash *EmployeeDetails:ID:1* is no longer needed.

Let's use the HDEL command to remove the salary field from the hash.

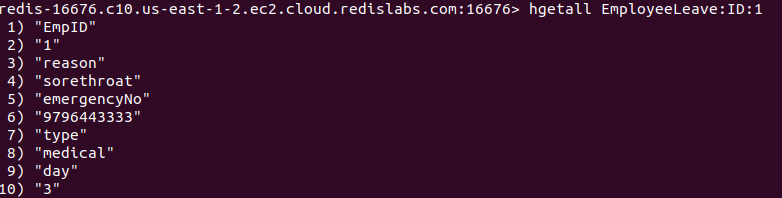
| hdel EmployeeDetails:ID:1 Salary |
| --- |

This should delete the field *salary* and its value.

Output:



Let's say the company has stored employee leave details in a separate hash stored at key *EmployeeLeave:ID:1.* This hash contains the following details about an employee.



Now the company thought of removing a few unnecessary fields from this hash. Since the leave reason and emergencyNo fields are no longer needed, we will use the HDEL command to remove those two fields at once.

| hdel EmployeeLeave:ID:1 reason emergencyNo |
| --- |

Output:



As expected, the command returned 2 that means two fields have been successfully deleted from the hash *EmployeeLeave:ID:1.*

Let’s try to remove a field that doesn’t exist in the hash.

| hdel EmployeeLeave:ID:1 field1 |
| --- |



It returned 0 which means nothing happened because field1 doesn’t exist.

Whenever you specify a hash key that doesn’t exist, the HDEL command will return 0 as shown in the following.



The HDEL command is a very simple command to use and it can be used to remove multiple field-value pairs from a hash efficiently.

# Conclusion

In summary, Redis hashes can hold an array of string field-value pairs in small memory. A hash can store more than four billion fields in a Redis instance. Hence, it is a very memory-optimized data structure to be used in real-world applications that deal with hundreds of objects. As discussed, whenever you need to remove one or more unnecessary field-value pair/s from a Redis hash, you can use the HDEL command. It operates in O(N) time complexity. Overall, the HDEL command is a very useful command to get rid of unwanted fields in a Redis hash stored in a given key.