Lesson 1: Tensor Index Notation

It’s now time to shift gears into something much more abstract than standard vector calculus.

At this point, we’ve built up the necessary knowledge to understand any physics topic applying vector calculus and it’s now time to expand upon this knowledge with the goal of giving you the necessary tools to understand much more advanced topics like **general relativity**.

We’ll begin with familiarizing ourselves with some of the notation used in **tensor calculus.** To put it simply, you can think of tensor calculus as a generalization of vector calculus - nearly everything we've talked about previously will apply, but in a more general and in a bit more of a difficult way.

We will be applying the tools of tensor calculus mostly to vectors at first as this is the easiest way to introduce the topic. In a few lessons from now, we'll then move on to study **tensors** (which can be thought of as generalizations of vectors).

Lesson Contents:

1. Index Notation For Vectors

The standard piece of notation you'll encounter countless in tensor calculus is **index notation**. This way of denoting stuff has a couple of advantages:

- Index notation allows us to deal with vectors (and tensors) directly in terms of their components instead of having to always express vectors either using vector notation or (explicitly) in terms of basis vectors.

- Index notation allows for sums of terms to be expressed neatly. In both vector and tensor calculus, summation operations are everywhere, so expressing them in a non-cluttered and simple way is extremely useful.

Now, what is this index notation? Well, it's simply just a different way of expressing the same thing we already know how to express - namely, **vectors and their components**.

1.1. Expressing Vectors In Terms of Their Components

As an example, let's consider the following vector:

$$\vec{v} = 3\hat{x} + 7\hat{y} + 2\hat{z}$$

For a vector like this, we really only need two pieces of information to express it in the above form; a **coordinate system** (i.e. a set of basis vectors) and the **vector components in this particular coordinate system**. If we know these two, we can specify everything there is to specify about the vector.

In this example, the coordinate system is the Cartesian system with basis vectors $\hat{x}$, $\hat{y}$ and $\hat{z}$. The components of this vector in the Cartesian system are 3, 7 and 2.

Using these two pieces of information, the vector can be written as a sum of the basis vectors and its components, which is what you see in the above example. In general, this way of writing a vector is called a *linear combination of the basis vectors.*

Now, we can write this linear combination in a general way as:

$$\vec{v} = \sum_i v^i \vec{e}_i$$

Here, v^i denotes the **vector components** with i being a summation index that runs from 1 to however many coordinates there are (in our example above, this would be 3). The $\vec{e}_i$ here are the **basis vectors** in each direction, so going by our above example, $\vec{e}_1$ would be the x-basis vector, $\vec{e}_2$ the y-basis vector and so on.

By writing the sum out, you get an expression of the form $v^1\hat{e}_1 + v^2\hat{e}_2 + v^3\hat{e}_3$. Note that i here in v^i is an index (upstairs index), not a power. We'll talk more about what that this means in an upcoming lesson.

Writing the linear combination as a compact summation as shown above is just a more general and more convenient way of expressing a vector. First, it's easier to label the different coordinate directions with an index i (where $i = 1, 2, 3...$) instead of a new letter for each.

Second, the above form doesn't make reference to any particular coordinate system - it's a much more general formula. The v^i and $\vec{e}_i$ could denote the vector components and basis vectors in Cartesian coordinates, but they could also refer to spherical coordinates or any other orthogonal coordinate system.

This is what we typically want to do in tensor calculus - work with expressions that are as general as possible and not just applicable in one particular coordinate system.

Now, what we really care about here are the vector components v^i. We can write these as a list of numbers (this is called a column vector; there also exists row vectors, but we're not going to need them now):

$$v^i = \begin{pmatrix} 3 \\ 7 \\ 2 \end{pmatrix} \begin{matrix} \mathbf{V^1} \\ \mathbf{V^2} \\ \mathbf{V^3} \end{matrix}$$

This way of writing a vector is called a matrix representation. However, you won't have to know about matrices to understand what we're going to talk about.

The interesting thing is that this way of writing the vector already specifies everything about the vector. We know that, for example, $v^1 = 3$ is the x-component of our vector, in other words, the component associated with the basis vector in the x-direction.

Therefore, in this kind of notation, the basis vectors are already "built in", meaning that we do not have to explicitly express them, we just have to know that the direction associated with the index $i = 1$ represents the x-direction and so on.

The point here is that it is possible to express a vector using only its components, v^i, however, we just have to keep in mind that the components here aren't *really* the vector itself, they just refer to the vector in a particular basis.

So, there are two types of equivalent ways to work with vectors:

- We can work directly with the vector $\vec{v}$ itself. The full vector is a geometric object that is the same in all coordinate systems. Any results we obtain like this are fully general and retain their forms in any coordinate system.
- We can work directly with the vector components v^i. The actual values of the components depend on the basis we're working in, so they are not the same in all coordinate systems. The results we obtain like this will in general have different forms in different coordinate systems, but there are ways in which we can predictably transform between different coordinate systems.

Now, it might seem like the first way is better. We want equations that are completely general and have the same form in all coordinate systems, right? Well, not necessarily.

The power of coordinate systems is exactly the fact that in different coordinate systems, things look different. Choosing the right coordinate system for a given situation is what we pretty much always do when we want to understand the physics of a system.

For example, consider Newton's second law, $\vec{F} = m\vec{a}$. This is a vector equation that (at least classically) has the same form in all coordinate systems. However, this equation is not particularly useful in this form - it has the same form for a projectile, a pendulum or anything else, so if we want to actually use $\vec{F} = m\vec{a}$ to solve a particular problem in practice, we have to specify a particular coordinate system.

If we do so, we can then express the force and acceleration vectors in terms of their components in that particular coordinate system and get useful physical results.

The point with all of this is that working directly with vector (and tensor) components - things with indices like v^i - is vastly more practical than having to carry around entire vectors in our equations. This is exactly what we do in tensor calculus.

Now, the central question that arises from this is "if we work directly with vector components and that requires us to pick a specific basis, doesn't that mean that all the results we obtain only apply in that particular basis?".

The answer is actually no. It turns out that while individual vector components can be different in different bases, the relationships or equations between these vector components retain their

form in all coordinate systems. This property is called *covariance*, which we'll discuss more soon.

This is really the power of tensor calculus - any equation we obtain in terms of vector components in one basis (that is covariant), we will automatically know in any other basis. So, with the use of index notation, we can express vector components just as generally as the vector itself without actually having to pick any specific basis to work in.

If we have a set of vector components v^i, the index i might refer to the different Cartesian coordinate directions (x, y, z) or it might refer to the spherical coordinate directions (r, θ, φ). The point is that we don't have to necessarily specify this in advanced, we can just say "here's some vector components v^i in some general basis with coordinate directions labeled by the index i".

However, when working with index notation, you should still keep at the back of your mind the distinction between vector components and a vector itself - strictly speaking, they are not the same thing (this distinction can be understood by the notions of invariance versus covariance). There are certain rules for using index notation that will become clear throughout our discussions on tensors.

2. Covariance vs Invariance

Let's briefly talk about the distinction between covariant and invariant quantities and equations. We usually refer to these terms when talking about equations or laws of physics.

Simply put, **an *invariant* equation is an equation that is exactly the same in all coordinate systems**. Just by the definition of the word, invariant means "not changing", which in physics refers to coordinate transformations.

For example, Newton's second law, $\vec{F} = m\vec{a}$, is an invariant equation (at least in classical mechanics) - it is constructed out of full vectors and scalars, which by definition, are invariant geometric objects that must be the same in all coordinate systems. So, $\vec{F} = m\vec{a}$ is still exactly $\vec{F} = m\vec{a}$ whether we are working in Cartesian, polar, spherical or any other coordinate system. Everything in the equation remains exactly the same in all coordinates, so we say the equation is invariant.

On the other hand, **a *covariant* equation is an equation that has the same *form* in all coordinate systems, but different quantities in the equation can be different in different coordinate systems**.

The word covariant means roughly "changing together", which captures the definition quite well. When an equation is covariant, the different quantities in the equation can change under coordinate transformations, but each of these quantities must change in the same way - "changing together" - as to make the equation itself retain its form.

For example, if we take Newton's second law and express it using vector components as $F^i = ma^i$, this is now a covariant equation - the vector components F^i and a^i do change under a coordinate transformation (unlike the vectors $\vec{F}$ and $\vec{a}$ themselves), but since these components appear on both sides of the equation and they transform in the same way (since they are vector components), the equation itself retains its form.

So, if we have the equation $F^i = ma^i$ in say, Cartesian coordinates, then in polar coordinates, the vector components F^i and a^i will be something different, say $\overline{F}^i$ and $\overline{a}^i$, but the equation itself will be $\overline{F}^i = m\overline{a}^i$ - while F^i and a^i individually can change, the equation itself retains its form.

Now, I haven't proven anything here yet, I've simply stated that this is what happens. We'll come back to why this is so when talking about coordinate transformations, but as a small "appetizer", we'll discover that the vector components transform by a coordinate transformation matrix $\Lambda^{\bar{i}}_j$ (called the Jacobian matrix) as $\overline{F}^i = \Lambda^{\bar{i}}_j F^j$ and $\overline{a}^i = \Lambda^{\bar{i}}_j a^j$, which means that the full equation, $F^i = ma^i$ transforms under a coordinate transformation to:

$$\begin{aligned}
&\Rightarrow \quad \overline{F}^i = m\overline{a}^i \\
&\Rightarrow \quad \Lambda^{\bar{i}}_j F^j = m\Lambda^{\bar{i}}_j a^j \\
&\Rightarrow \quad F^j = ma^j
\end{aligned}$$

The equation retains its form! Roughly speaking, you can think of this as resulting from the fact that since both sides of the equation transform the same way, the transformations "cancel out" on both sides, leaving the form of the equation the same.

The key to understand here is that while $\overline{F}^i - m\overline{a}^i = F^i - ma^i$ under any coordinate transformation (this is what it means for the equation to be covariant), this does NOT say that $\overline{F}^i = F^i$ or $\overline{a}^i = a^i$. In other words, the individual quantities, the vector components, are different in different coordinate systems, but the *relationship* between the different components stays the same.

However, in an invariant equation, we have $\vec{F} - m\vec{a} = \vec{F} - m\vec{a}$ and the individual quantities, vectors in this case, ARE also the same, so $\vec{F} = \vec{F}$ and $\vec{a} = \vec{a}$. So, in an invariant equation,

everything stays exactly the same, while in a covariant equation, only the relationship between quantities - but not the quantities themselves - stays the same. Invariance is a much stricter condition than covariance.

$$F^i = ma^i \quad \xrightarrow{\text{Coordinate transformation}} \quad \bar{F}^i = m\bar{a}^i$$

When an equation is *covariant*, both sides of the equation transform in the same, well-predictable way under coordinate transformations and this leaves the *form* – the relationship between the two sides of the equation - the same

$$\vec{F} = m\vec{a} \quad \xrightarrow{\text{Coordinate transformation}} \quad \vec{F} = m\vec{a}$$

When an equation is *invariant*, both sides of the equation remain exactly the same under coordinate transformations and this leaves the entire equation – including every quantity in the equation - exactly the same

But why is any of this important? Well, in physics, we require the laws of physics to be **covariant** (or invariant, but covariance is more general). Any valid law of physics that applies to all systems in nature (within the framework of a given theory) needs to be built out of quantities that transform in a way in which the equation describing the law stays covariant.

Physically, the reason for this is that if an equation is taken as a law of nature - in other words, it describes a fundamental relationship between physical quantities - then this relationship between the quantities better hold in ALL coordinate systems, since changing coordinate systems should no change the actual physics we're describing.

Coordinate systems are just our way of quantitatively describing physical systems and making predictions about what will happen, but the coordinates themselves don't exist in nature.

However, of course, if we want to analyze a specific physical system in a specific coordinate system, this is when we need to specify a particular coordinate basis and obtain results in that particular basis.

The important thing is that while a particular description of a physical system - such as the numerical values of, say a velocity - can be different in different coordinate systems, the physical laws and fundamental relationships from which these coordinate descriptions are obtained need to be independent of any particular coordinate system, so either invariant or covariant. This is a general theme that will come up time and time again in, for example,

general relativity.

Now, as it turns out, the covariant objects we can build our covariant equations and laws of nature out of are exactly the scalars, vector components and more generally, tensors we've been talking about here. This is why we need tensor calculus in physics.

The key takeaway with all of this is to understand *why* we use this index notation and work directly with vector and tensor components and why it all works.

The reason is that working with indices and components directly actually turns out to be quite simple and straightforward once you learn all the rules and get the hang of it. And most importantly, the reason it works is because of the transformation properties of vector and tensor components, which results in any equations constructed out of these to be covariant - exactly what we want in physics!

Sidenote: While it is true that any vector equation in terms of vector components will be covariant in general, what constitutes a valid vector can actually be different in different theories of physics.

For example, in Newtonian mechanics, the three-dimensional acceleration and force vectors with components a^i and F^i are valid vectors, so an equation of the form $F^i = ma^i$ is a covariant equation. However, in relativistic theories with a four-dimensional spacetime, it turns out that the three-dimensional acceleration and force vectors are actually not valid vectors, so an equation of the form $F^i = ma^i$ is not relativistically covariant. In relativity, we instead consider objects called four-vectors, which are valid vectors in four-dimensional spacetime as well and all relativistic theories need to be formulated in terms of these.

More specifically, what defines a valid vector in a particular theory is how the components of the vector transform under the action of the symmetry group of that particular theory. For example, in Newtonian mechanics, this symmetry group is the Galilean group and any vector whose components transform in a specifically defined predictable way (we will come back to how exactly in the next lesson) under the action of the Galilean group is taken to be a valid vector in Newtonian mechanics. In special relativity, on the other hand, this symmetry group would be the Poincaré group and any vector whose components transform in a "predictable way" under the action of the Poincaré group is taken to be a valid relativistic vector (called a four-vector). Again, we'll come back to what this "predictable way" means exactly in the next lesson.

3. The Einstein Summation Convention

Okay, we've now established why the index notation and working directly with vector components works and why it's useful.

From now on, we will use this index notation to express vectors - or more accurately, vector components - by an upstairs index or as we'll come to see, a downstairs index, which denotes something called a covector.

So, anytime you see a thing with **one index**, this indicates that it is a vector. From now on, I'll be using the words vector and vector components somewhat interchangeably since this is what you'll commonly see in the literature. However, strictly speaking, you should keep in mind that a thing with indices - like v^i - refers to vector components, while a thing with an arrow - like $\vec{v}$ - refers to the *actual* vector.

It's also worth noting that objects with no indices typically denote scalars. The important thing about scalars is that they are invariant under coordinate transformations.

Now, the next thing we'll do is dive deeper into *how* our index notation actually works in practice. One of the most important and commonly used rules is the **Einstein summation convention**, which will greatly simplify most calculations we do using this index notation.

We'll begin by going back to our earlier definition of a vector expressed as a linear combination of the components with the basis vectors:

$$\vec{v} = \sum_i v^i \vec{e}_i$$

Can you notice something here? The vector components are written with an *upstairs index*, while the basis vectors have a *downstairs index* (the names for these are contravariant and covariant indices, or in more modern terminology, vector and covector indices).

The nice thing about this, which I suppose Einstein noticed first when doing calculations in general relativity, is that **whenever we have happen to have a sum over an index, the index appears in a term that has both an upstairs and a downstairs index**.

In other words, any time you see a term that has the same index in both an upstairs and downstairs position, there will be a summation over that index.

Since this is such a general feature that occurs when working in this index notation, the standard convention is to just leave out the summation sign and write the above thing as:

$$\vec{v} = v^i \vec{e}_i$$

As you can see, this is a much cleaner way to write things. This convention of leaving out summation signs whenever there are both an upstairs and a downstairs index repeated in the same term goes by the name of the *Einstein summation convention*.

Now, you may ask what the point of this really is. Doesn't it just make things more difficult since we have to then remember where all these "implicit" summations are in our equations?

Well, not really, it's just something you need to get used to. You just have to remember this one rule: **whenever there is a repeated index in both the upstairs and downstairs position in the same term, we implicitly sum over them**. That's it.

This convention is actually quite useful for simplifiying our equations and if you just remember the rule stated above, you shouldn't run into any trouble. As an example, in tensor calculus, you might come across expressions like this:

$$\sum_i \sum_j \sum_m T^{ijm} V_{ijmn} + \sum_s \sum_t \sum_k U^{stk} K_{stkn}$$

These things with multiple indices are called **tensors**, but more on those later.

Now, imagine doing calculations having to carry around all these summation signs. You'd probably go crazy and very likely make a mistake by accidentally leaving one of them out or something like that. A much less cluttered way would be to use the Einstein summation convention and just write this as:

$$T^{ijm} V_{ijmn} + U^{stk} K_{stkn}$$

This is much cleaner and if you just remember the rule of repeated upstairs and downstairs indices being implicity summed over, you will automatically remember which indices should be summed over and which ones not.

Anyway, the Einstein sum convention together the vector index notation allow us to also write common expressions in a neat form. For example, the dot product between two vectors, $\vec{v}$ and $\vec{u}$, would be written as:

$$\vec{v} \cdot \vec{u} = v^i u_i$$

Here, we need to write one of the vectors with a downstairs index for the Einstein sum convention to be valid. We'll talk more about what these things with downstairs indices are in later lessons, but in Cartesian coordinates, the vector components u^i and u_i are the same thing.

We can also write the cross product components using the Levi-Civita symbol as (we covered this in the lesson Coordinates, Vectors & Basis Vectors):

$$(\vec{v} \times \vec{u})_i = \varepsilon_{ijk} v^j u^k$$

Here, both j and k are summed over as dictated by Einstein's summation convention. Notice that on the left, we have a vector $(\vec{v} \times \vec{u})_i$ that has a downstairs index, which is simply due to the fact the i-index on the right is also in the downstairs position and they should be the same on both sides of the equation.

The thing on the left is therefore a dual vector (a thing with a lower index) as opposed to being the same kind of vector we often think of the cross product as being. We'll come to back to these a bit later but for doing practical calculations, the distinction between vectors and dual vectors is really not that important since it turns out that these indices can be "raised" and "lowered" in a straightforward manner.

Now, the key point is that this index notation and the Einstein sum convention provide a nice and simple way to express vectors and do vector operations in (and tensor operations as well). Together, these provide a foundation for the notation we're going to be using from now on.

I do, however, also understand that there is a lot of getting used to with these new notational tools. But as with anything, all it requires is just some time and examples.

4. Example: Differential Operators Using Index Notation

Speaking of examples, let's see how the different differential operators we've encountered in this course look like in our newfound index notation. We'll write the gradient, directional derivative, divergence, curl and Laplacian using this notation.

Note that these expressions are only going to be valid in Cartesian coordinates. We will get to how these can be generalized to any coordinates in a later lesson. The point of this example is to just get us familiar with using this new index notation.

Let's begin with the **gradient** of a scalar field:

$$\vec{\nabla} f = \frac{\partial f}{\partial x}\hat{x} + \frac{\partial f}{\partial y}\hat{y} + \frac{\partial f}{\partial z}\hat{z}$$

Here we have a sum of partial derivatives and basis vectors for each coordinate. Now, remember what we talked about before; the goal is to express vectors using indices in the

“component form”, so in our notation, we would write a vector $\vec{v} = v^1\hat{x} + v^2\hat{y} + v^3\hat{z}$ simply as v^i with the index i denoting which component we're referring to.

We can do the same thing to the gradient expression above (since the gradient of a scalar field is a vector field - it has components just like any other vector). All we need to do is express these partial derivatives using indices, which is quite straightforward.

First, let’s define our set of coordinates as $x^i = (x, y, z)$. Note that these do not form an actual vector, this is just a list of coordinates, but we can still apply our index notation to it.

The partial derivative is the derivative with respect to each coordinate, which we can nicely express as $\partial / \partial x^i$. The gradient of f is then in component form:

$$\vec{\nabla} f \quad \Rightarrow \quad \frac{\partial f}{\partial x^i}$$

Now, this expression is our gradient vector (again, in Cartesian coordinates) in component form using index notation. It's as simple as that! Well, actually, it's not *quite* that simple. The "issue" here is that the expression $\partial f / \partial x^i$ is, to be precise, actually a thing with downstairs indices.

So, even though the partial derivative operator ($\partial / \partial x^i$) involves an upstairs index thing (x^i), the partial derivative operator itself is a downstairs vector. A more clear way to express this would be as follows:

$$\frac{\partial}{\partial x^i} = \partial_i$$

Now, intuitively you could think of this in the following way; since the coordinates x^i with an upstairs index are “below” the fraction bar, this makes the whole thing actually something with a downstairs index.

The real mathematical reason for this, however, is that the partial derivatives transform as components of a “covector” (a vector with downstairs indices), which we'll talk more about later. So, it’s actually correct to write the partial derivative with the index in the opposite position as in the coordinates (so, if we label the coordinates as x^i, the partial derivative should be ∂_i - coordinates are always written with an upstairs index, so there is no such thing as x_i!).

Now, since the partial derivatives are things with downstairs indices, we cannot write the full gradient vector as a linear combination with the basis vectors simply as:

$$\vec{\nabla} f \neq \partial_i f \vec{e}_i$$

This is not a correct way to express this since we have two downstairs indices, which doesn't imply a summation anymore. The correct expression for the full gradient vector (in Cartesian coordinates) would be:

$\vec{\nabla} f = \delta^{ij} \partial_i f \vec{e}_j$, where $\delta^{ij} = 1$ for $i = j$ and zero for $i \neq j$.

This δ^{ij} is called the Kronecker delta or perhaps more accurately, the Euclidean metric, which is actually just an identity matrix. However, in different coordinate systems, this gets replaced by the metric tensor in that coordinate system (more on this in a few lessons).

With this, the "correct" gradient vector components would actually be $\delta^{ij} \partial_i f$ and this would indeed be an expression with upstairs indices (meaning this is a vector like we expect the gradient to be) - the partial derivatives $\partial_i f$ themselves have downstairs indices and we call would call it a covector instead. More on these later.

However, in Cartesian coordinates, the nice thing is that there actually is no distinction between upstairs and downstairs indices and because of this, we can leave the δ^{ij} out completely - it makes no difference. We'll look at the more general definition of a gradient later, but for now, we'll just take the gradient in component form to be $\partial_i f$.

We can now use this result for the other differential operators as well. Let's do the **divergence** next. This is simply the dot product between a vector field and the gradient operator.

Now, remember from earlier where we concluded that we could express a dot product using inex notation and Einstein's sum convention as (remember that i is being summed over here!):

$$\vec{v} \cdot \vec{u} = v^i u_i$$

The divergence would therefore be:

$$\vec{\nabla} \cdot \vec{f} = \partial_i f^i$$

We can see that this expression gives the correct result by writing out the sum here and checking what we get:

$$\partial_i f^i = \partial_1 f^1 + \partial_2 f^2 + \partial_3 f^3 = \frac{\partial f^x}{\partial x} + \frac{\partial f^y}{\partial y} + \frac{\partial f^z}{\partial z}$$

As a reminder here, in Cartesian coordinates, the indices $i = 1, 2, 3$ refer to x, y, z and therefore $\partial_1 = \partial / \partial x$ and so on. The point is that we indeed get the formula for the divergence (in the Cartesian basis) as we should.

Using the same logic, which I hope you get the idea of, we can write the **directional derivative** in index notation as well. The directional derivative would simply be the dot product between the gradient of a scalar field and some vector $\vec{v}$, which using index notation, looks as follows:

$$\vec{\nabla}_{\vec{v}} f = \vec{v} \cdot \vec{\nabla} f = v^i \partial_i f$$

Now, let's do the **curl** next. For this, we just need the formula for the cross product in index notation:

$$(\vec{v} \times \vec{u})_i = \varepsilon_{ijk} v^j u^k$$

All we have to do is replace $\vec{v}$ with the gradient operator and $\vec{u}$ with the vector field $\vec{f}$ (just to keep things consistent):

$$(\vec{\nabla} \times \vec{f})_i = \varepsilon_{ijk} \partial^j f^k$$

Note that we now have the partial derivative with an upstairs index here. This would be the operator defined as $\partial^i = \delta^{ij} \partial_j$. In general, the placement of indices DOES matter, but in the simple case of Cartesian coordinates, we have $\partial^i = \partial_i$. This is going to make more sense when we get to raising and lowering indices using a metric.

However, it is also possible to write the curl with an upstairs index as:

$$(\vec{\nabla} \times \vec{f})^i = \varepsilon^{ijk} \partial_j f_k$$

Now, the last operator we'll write using this notation is the **Laplacian**. The Laplacian is actually quite simple. Let's first write the dot product between gradient operators using the Einstein sum convention as:

$$\nabla^2 = \vec{\nabla} \cdot \vec{\nabla} = \partial_i \partial^i$$

Then, the Laplacian of a scalar field is simply:

$$\nabla^2 f = \partial_i \partial^i f$$

If all of these things seem overwhelming, don't worry. Things should make more sense as we

get further into index notation. We'll also learn later on how to change between upstairs and downstairs indices as well as all kinds of useful index manipulation techniques.

For the sake of comparison, I've collected the differential operators discussed here into a table down below, where you'll see these written using both the standard vector notation and the new index/component form notation (in Cartesian coordinates).

Differential operator	Standard vector notation	Index notation
Gradient of a scalar field	$\vec{\nabla} f$	$\partial_i f$
Directional derivative of a scalar field	$\vec{\nabla}_{\vec{v}} f$	$v^i \partial_i f$
Divergence of a vector field	$\vec{\nabla} \cdot \vec{f}$	$\partial_i f^i$
Curl of a vector field	$\vec{\nabla} \times \vec{f}$	$\varepsilon_{ijk} \partial^j f^k$
Laplacian of a scalar field	$\nabla^2 f$	$\partial_i \partial^i f$

An important point to note with these again is that the index notation formulas are only valid in the Cartesian basis - in different coordinate bases, the operators will be a bit different. We'll derive the general formulas valid in all coordinate systems for these in a later lesson.